Player Achievements Technical Reference

Feature Overview

The Achievements system in Horizon Worlds makes it easy to define a list of player-objectives, persistently track which ones a person has accomplished across multiple visits to your world, and communicate progress to players. You define achievements, read and write them with code blocks, and can show them to people in a specific achievements gizmo.

This system is separate from, but interoperable with, the Player Persistent Variable and Leaderboard systems. Think of it as a complementary mechanism for adding reasons to return to your world by providing hints to players about what is possible in your world, and rewards for when they achieve certain objectives.

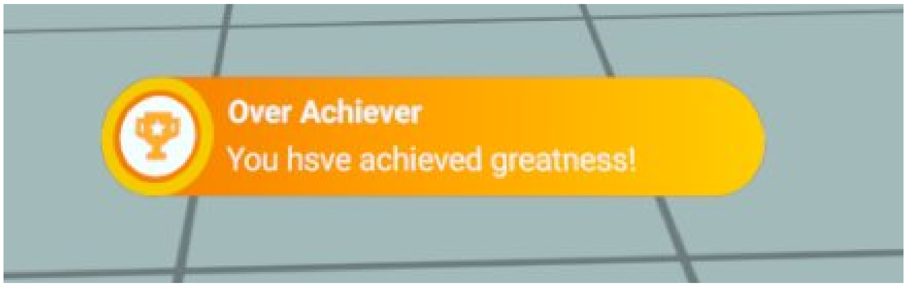
Achievements are persistent boolean flags tracked independently for each player. Once an achievement is marked complete, it remains complete unless it is explicitly marked incomplete.

There are two kinds of achievements, described in further detail below:

- Simple - marked complete by a specific code block you execute
- Tracked Persistent - marked complete automatically when a tracked Player Persistent Variable (PPV) reaches a pre-configured threshold value.

Achievements are automatically displayed to the player via two mechanisms: an Achievements Panel gizmo object placed in the world, and by short lived popups that appear in front of the player when the achievement is completed. The popup has nearly identical format to the completed line of the achievement in the Achievements Panel gizmo. You can use the code blocks below to create additional custom displays of achievement progress or status.

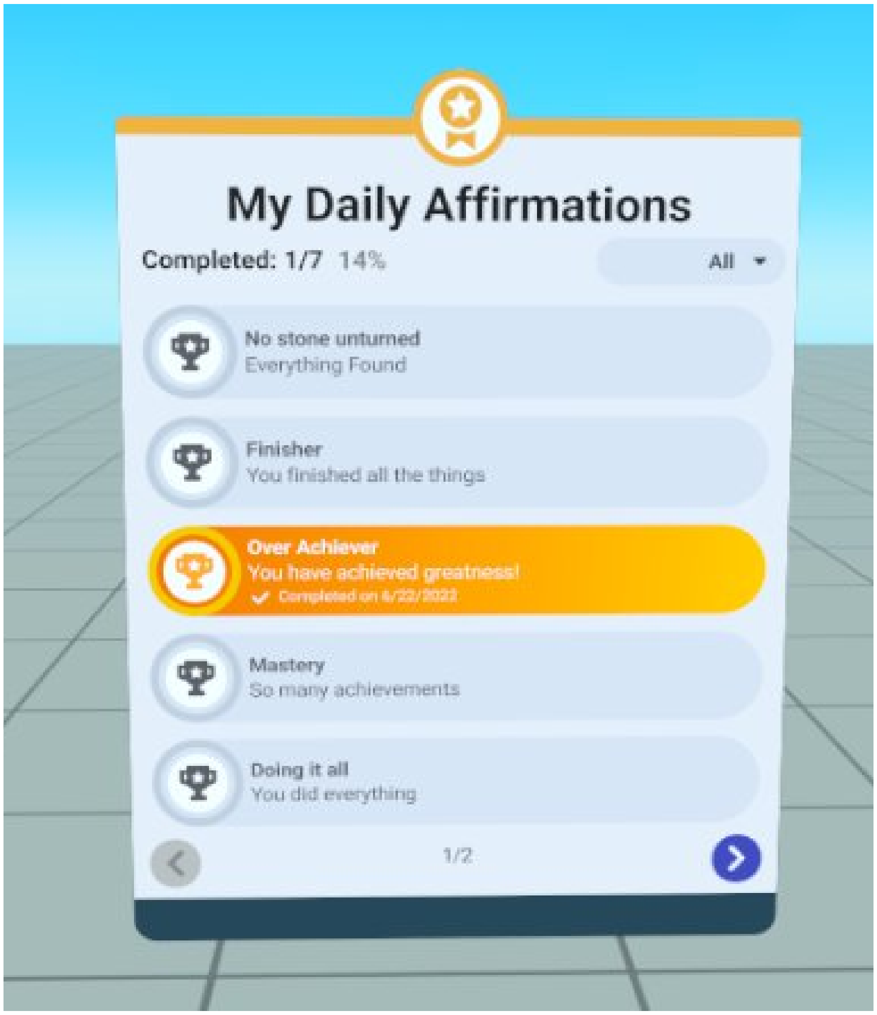
An achievements popup


Over Achiever

You hsve achieved greatness!

Gizmos

Achievements Panel



My Daily Affirmations

Completed: 1/7 14%

No stone unturned
Everything Found

Finisher
You finished all the things

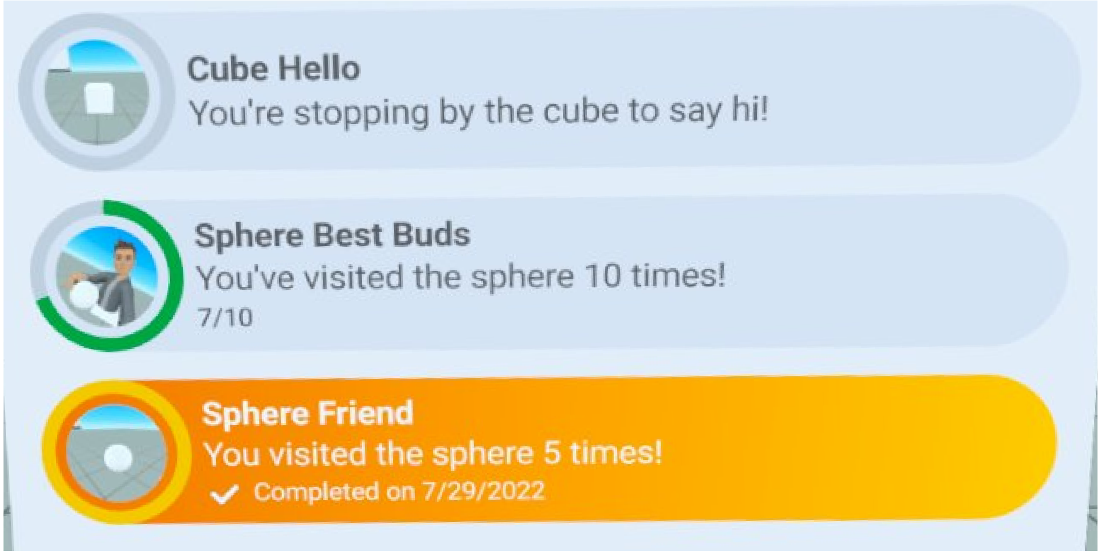
Over Achiever
You have achieved greatness!
✓ Completed on 6/22/2022

Mastery
So many achievements

Doing it all
You did everything

The Achievements Panel Gizmo (default object name when pulled from the gizmo menu is “GoalsPanel”) is the primary UI element of this feature. It is very similar in appearance to the Leaderboard Gizmo, with the ability to flip through multiple pages of achievements. Rather than a tabbed view, the player can use a a drop down filter menu to restrict the display to “All”, “In Progress” (i.e. incomplete) or “Completed” achievements.

Each achievement is displayed, one per line, with the following information:



Cube Hello
You're stopping by the cube to say hi!

Sphere Best Buds
You've visited the sphere 10 times!
7/10

Sphere Friend
You visited the sphere 5 times!
✓ Completed on 7/29/2022

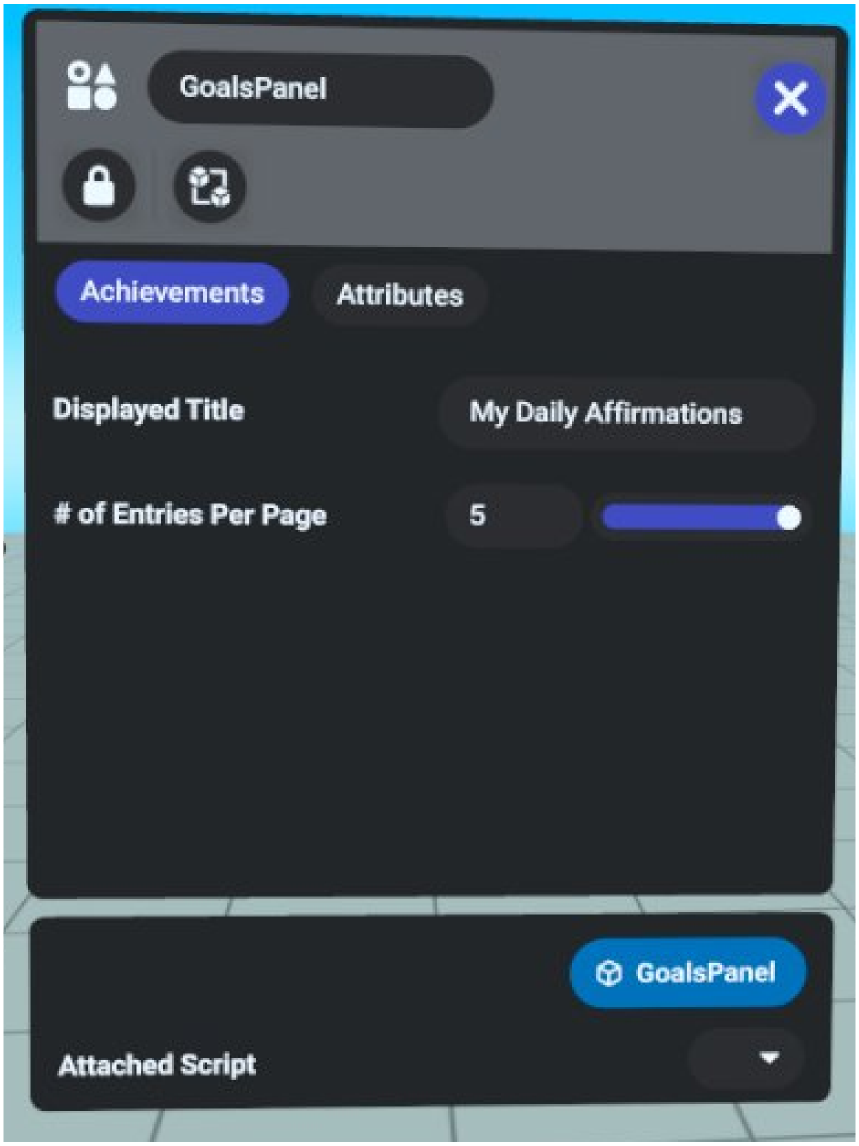
- Thumbnail: An optional image that identifies the achievement
- Progress: A ring around the thumbnail that indicates how close to completion this achievement is
- Name: A bold line of text with the achievement name
- Description: A smaller line of text with a more detailed description of the achievement
- Completion: Either absent (if unstarted), the fraction complete (if partially complete), or the completion date

Incomplete achievements are light gray, while completed achievements are gold colored.

The Achievement Panel provides a ‘player specific’ view of Achievements. Multiple players can be looking at the same Achievements Panel gizmo, but each will see only their own achievement progress. Switching pages or changing the drop down filter also only affects the view of the player making the change.

Object Properties

Unlike the Leaderboard, there is no need to set an object property on the Achievements Panel to connect it up to a pre-configured system setting. All defined achievements in the system (see below) are available for display in any Achievements gizmo (however, see below on how to use scripts to set up per-panel filters).



Displayed Title My Daily Affirmations

# of Entries Per Page 5

GoalsPanel

Attached Script

The panel has a few configurable object properties, again similar to Leaderboards.

Displayed Title

This is the text string that will be displayed to the player at the top of the Achievement Panel UI. Default value is “Achievements”

# of Entries Per Page

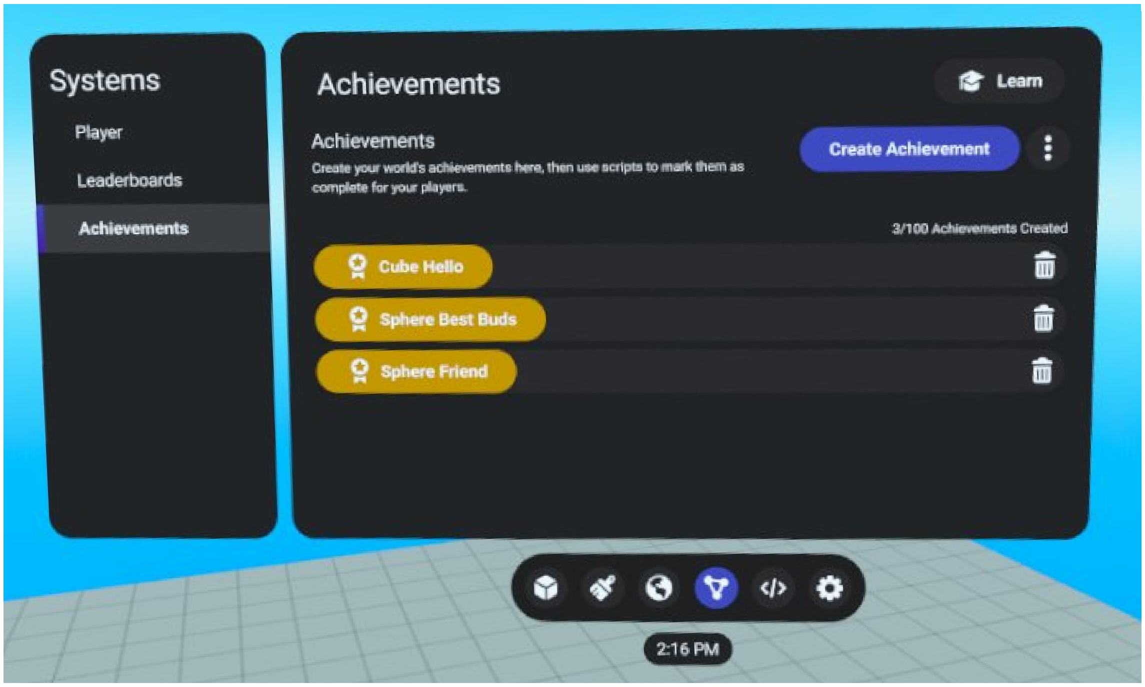
This is the number of achievements that will be listed per page in the UI. If your panel can display more achievements, the remainder will be available to the player via the next and previous page arrows at the bottom of the UI. Default value is 5.

Settings Panes

Configuration of achievements is done via the Build Settings->Systems->Achievements pane. There is some interaction with the Player Persistent Variable settings, explained below. Note that changes to Achievements can only be made by the Owner of the world. If an Editor makes changes to Achievements, they will not be persisted to the saved state of the world.

Systems->Achievements

The Achievements Settings pane displays all existing achievements configured for the current world. At present, you can create up to 100 different achievements per world. Each achievement can be edited after creation by clicking on its name. An achievement can be deleted by clicking on the trash can icon to the right of its name. Deletion of an achievement is permanent, and cannot be undone. The order of achievements in this list defines their default order in the Achievements Panel gizmo. Their order cannot be changed in settings, but see below for a scripted way to change their display order.



Create your world's achievements here, then use scripts to mark them as complete for your players.

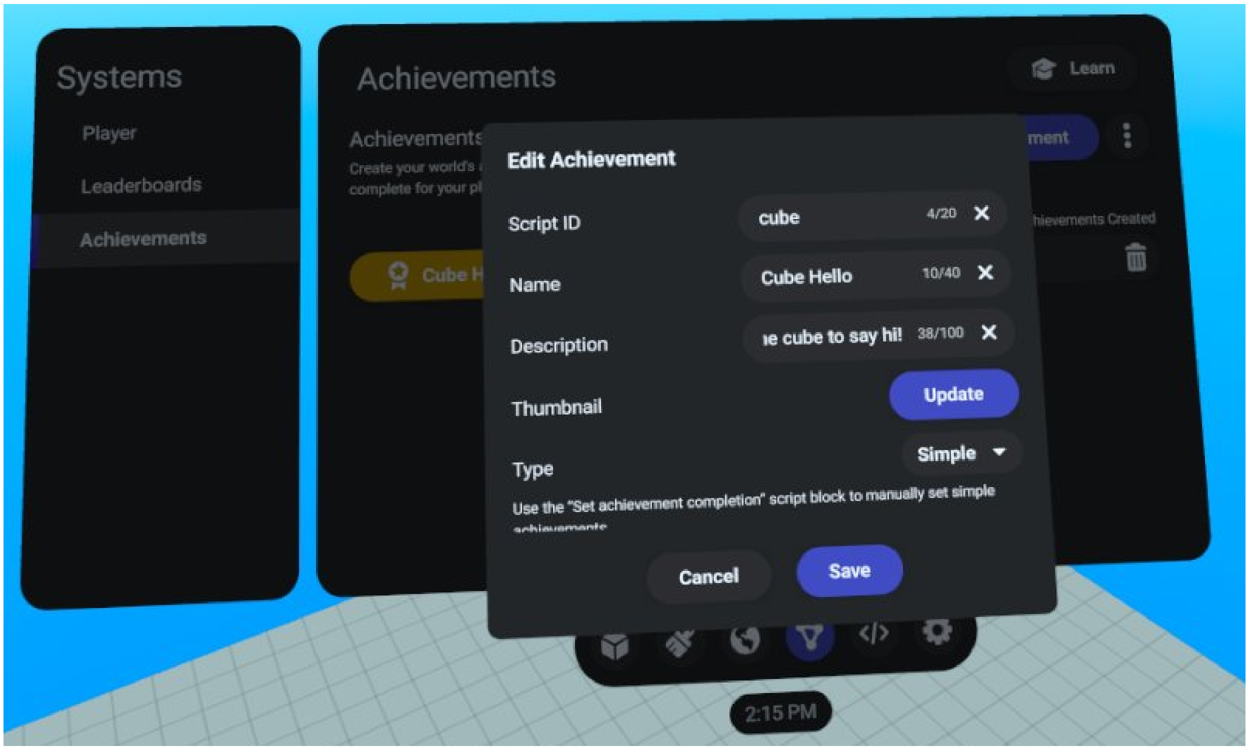
3/100 Achievements Created

There are two kinds of achievements that world Owners can create.

Simple

A simple achievement is a simple boolean value, and is either complete or it is unstarted. Simple achievements are marked complete via execution of a specific code block, described below. They can also be marked incomplete via code as well.

When configuring a Simple achievement, you must provide the following information:



Edit Achievement

Script ID: cube 4/20

Name: Cube Hello 10/40

Description: ıe cube to say hi! 38/100

Thumbnail: Update

Type: Simple

Use the "Set achievement completion" script block to manually set simple achievements

Script ID

This is an identifier used by some code blocks to identify this achievement via string fields. Note that not all code blocks use the Script ID, some use the Name in a drop down list. Note also that this ID has nothing to do with the name of the Script gizmos that will reference the achievement.

Generally speaking, this ID should be unique across all achievements in the world so that each achievement can be individually identified by scripts. However, if you use the same ID for multiple achievements, they can be addressed as a group by scripts. But, you will also no longer be able to differentiate them in scripts either.

Name

This is the name of the achievement that will be displayed to the player using bold text on the top line of the UI. Also, used in some code blocks to select the achievement via a dropdown list. It should be short but descriptive, i.e. a few words or a phrase.

Description

This is a longer block of text that describes what the achievement is for. It might provide hints to the player on how to complete the achievement, or other info for the player about why this achievement is desired. This is displayed in small text below the Name of the achievement.

Thumbnail

This is an optional image that is displayed to the left of the achievement in the UI. If not provided, a trophy image is displayed. Once set, there is no way to clear the thumbnail image, it can only be replaced. The thumbnail can only be chosen from images you have taken using your selfie stick, and are NOT restricted to images taken in the current world.

Type

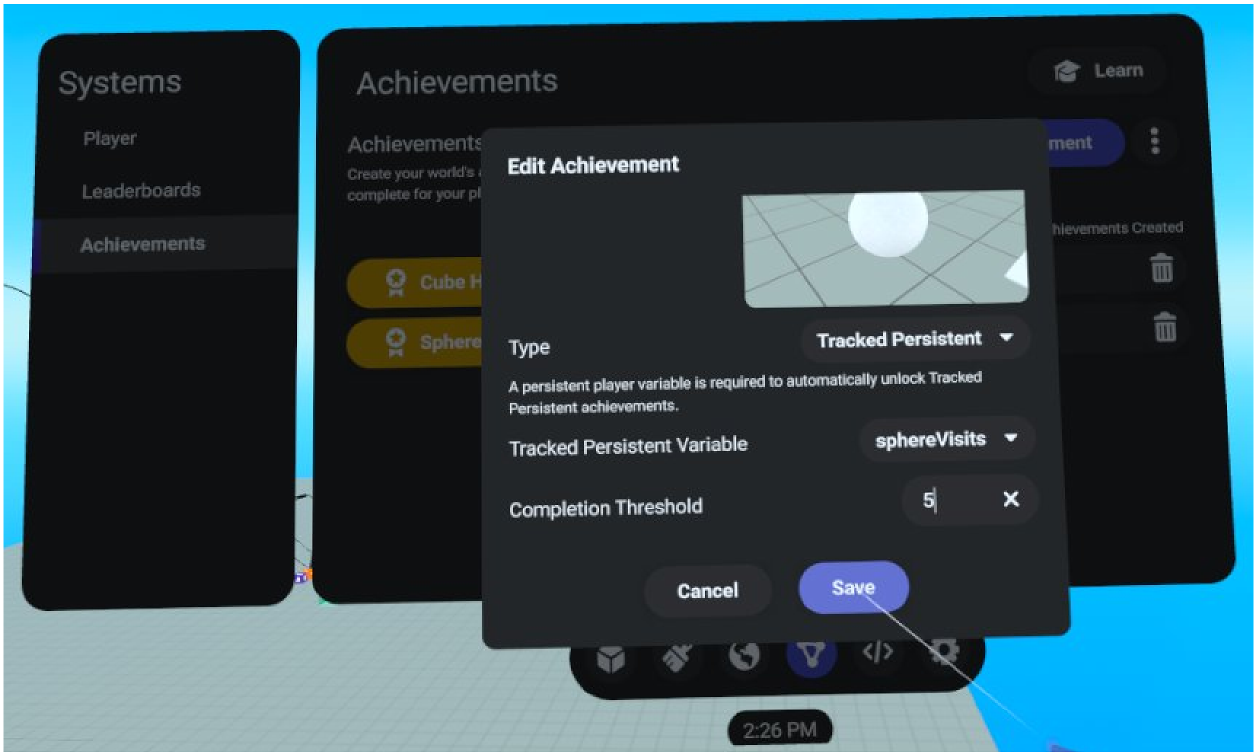
Simple Achievements use the “Simple” type.

Tracked Persistent

A Tracked Persistent achievement has all the capabilities of a Simple achievement, plus the ability to be marked complete *automatically* by the runtime when a tracked Player Persistent Variable (PPV) reaches a pre-configured threshold. Note that there is no mechanism to automatically mark Tracked Persistent achievements incomplete if the PPV later drops below this threshold. Completion via tracked PPVs is a ‘one way’ trip. Like Simple achievements, Tracked Persistent achievements can *also* be marked complete/incomplete via explicit code blocks.

Since Tracked Persistent achievements have a PPV attached, they can also be in a partially completed state. The UI uses this information to provide ‘fraction completed’ information to the player in the Achievements Panel. The ring around the thumbnail image is colored in proportional to the progress of the tracked PPV to the threshold value. The Completion line in the UI display contains the current value of the PPV relative to the threshold value (e.g. “4/15” for a PPV with the current value of 4, and a configured threshold of 15).

Tracked Persistent achievements have all the configuration parameters of Simple achievements described above (Script ID, Name, Description, Thumbnail), plus the following.



Edit Achievement

Type: Tracked Persistent

A persistent player variable is required to automatically unlock Tracked Persistent achievements.

Tracked Persistent Variable: sphereVisits

Completion Threshold: 5

Type

Tracked Persistent Achievements must use the “Tracked Persistent” type. Changing to this type will cause the following two additional fields to be displayed.

Tracked Persistent Variable

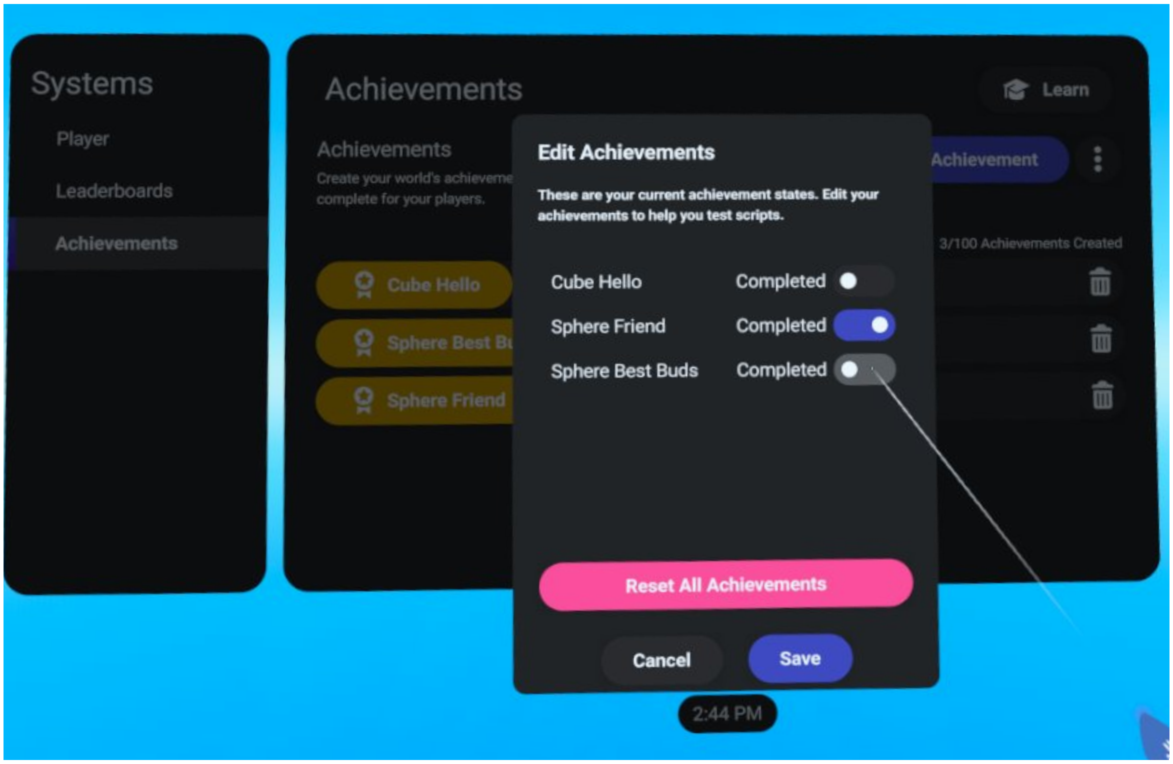
This is a drop down to select the *existing* PPV that will be tracked by this achievement. As noted above, you must set up this PPV *before* you create the Tracked Persistent achievement.

Completion Threshold

This is the threshold value for the PPV at which you want this achievement to be automatically marked complete. As noted above, if the PPV later drops below this threshold, the achievement *will not* automatically be marked incomplete.

Edit/Reset My Achievements

The triple-dot drop down menu in the upper right of the Achievements Pane allows you to edit the completion state of individual achievements, or reset all achievements at once to the incomplete state. This can be used during testing and debugging your use of achievements by either ‘rewinding’ the world to an earlier state to test the actions needed to complete achievement, or to ‘fast forward’ the world past the completion of certain achievements (without having to go through whatever tedious steps are necessary to do so) to debug the use of those achievements in their completed states.



Edit Achievements

These are your current achievement states. Edit your achievements to help you test scripts.

Cube Hello Completed ○
Sphere Friend Completed ●
Sphere Best Buds Completed ○

Note that while setting Simple achievements to complete/incomplete is straightforward and works as expected, Tracked Persistent achievements can have unexpected interactions with their corresponding PPVs. It is always possible to mark a Tracked Persistent achievement ‘complete’ via this UI. However, you cannot mark a Tracked Persistent achievement incomplete if the underlying PPV has a value that meets the configured threshold for the achievement. To mark such Tracked Persistent achievements incomplete, you must *first* set the PPV to a value below the threshold for the achievement. Note that you *can* reset a Tracked Persistent achievement to be in a partially completed state by setting the PPV to a nonzero value below the threshold, and then setting the achievement as incomplete.

Systems->Player

As described above, the Tracked Persistent achievements are configured to implicitly track the value of an existing Player Persistent Variable (PPV). The PPV must be created in the Settings->Systems->Player panel prior to creating the Achievement. See the “Player Persistent Variables” reference for more details on how to create/use PPVs.

Note that when resetting the PPV being tracked by a Tracked Persistent achievement below the threshold configured for the achievement, this *DOES NOT* mark the achievement incomplete! Achievements are only marked incomplete when explicitly set incomplete, either by a code block, or using the “Edit/Reset My Achievements” triple-dot dropdown in the Systems->Achievements panel.

Related Object Property Fields

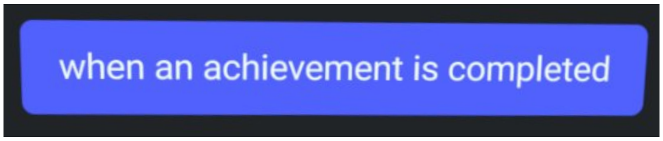
This feature proves no new Object Property fields on other existing kinds of gizmo/geometry objects.

Related Code blocks

This feature interacts with the following code blocks, grouped by their location in the script editor window and listed by their name in the code block picker pane. Their full code block syntax when placed in a script is described in detail below.

Events

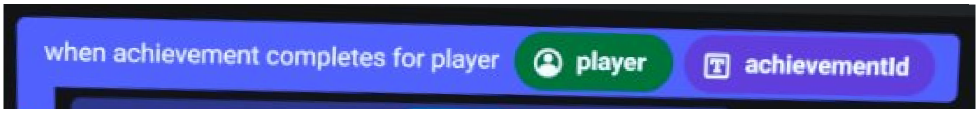
when an achievement is completed



when an achievement is completed

This event is delivered when any achievement is completed by any player. This event is useful when you want to take immediate action of some sort when a player completes an achievement, such as modifying a PPV for a player, attaching objects to the player, teleporting the player, or making some other non player specific change to your world when any player completes certain achievements.

When placed in a script, the full syntax of the code block is shown below



when achievement completes for player player achievementId

The parameters of the code block are as follows:

player

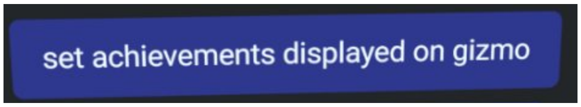
This is the identifier for the player that completed the achievement.

achievementId

This is the previously configured Script ID string of the achievement, and allows you to identify which achievement, exactly, was just completed.

Actions

set achievements displayed on gizmo

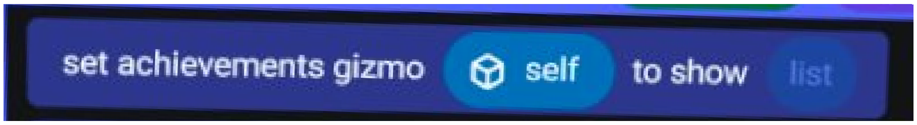


set achievements displayed on gizmo

This code block allows you to limit the achievements that are visible in a specific Achievements Panel gizmo, as well as the order of the achievements that are displayed. Each gizmo can have a different list of achievements configured for display. This configuration is global to all players, i.e. there is no way to set achievement ordering or visibility on a player by player basis.

You can use this code block to group achievements into multiple different panels for organization purposes (especially useful if you have a lot of achievements). Also, you can use this feature to hide certain achievements from view so that the player doesn't know they exist until they complete them. Note that completed achievements do not appear on the configured panel if they are not in the provided list. There is no way to make an achievement unlisted before completion, but then listed after completion.

When placed in a script, the full syntax of the code block is show below:



set achievements gizmo self to show list

Parameters of the code block are as follows:

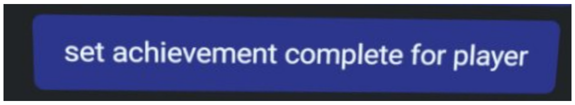
object

This is the achievement panel object on which you want to filter and reorder displayed achievements.

list

A list of strings containing the Script IDs of the achievements you want to be displayed in the provided Achievement Panel object reference. The order of the achievement Script IDs in the list defines the order they will be displayed in the panel gizmo. Any achievements with Script IDs *not* provided in the list will *not* be displayed in the panel.

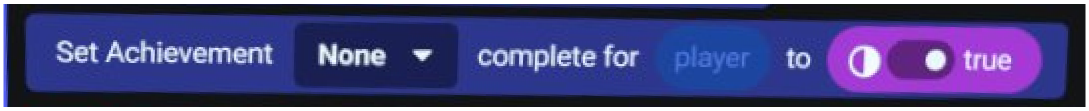
set achievement complete for player



set achievement complete for player

This code block is used to mark an achievement complete or incomplete, regardless of type. Note that while Tracked Persistent achievements are auto completed by the runtime when they reach their PPV threshold, they can also be marked complete explicitly.

When placed in a script, the full syntax of the code block is show below:



Set Achievement None ▾ complete for player to ◐ true

Parameters of the code block are as follows:

achievement

This is a drop down list of the Name values of all configured achievements. It is *not* the Script ID. Select the Name of the achievement whose state you want to change.

player

You must provide a reference to a player for which you want to update an achievement value. This player object can come from an existing variable, or via a parameter to an event handler (like trigger entry or collision).

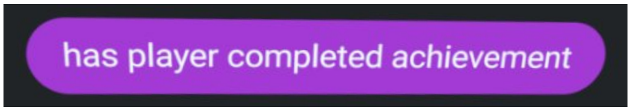
value

The new completion value for the achievement. If the achievement already has this value, nothing happens. If the achievement is changing to complete (i.e. true), a popup will appear in front of the player indicating completion. If changing completion state, any Achievement Panel gizmos configured to display the achievement will update its display accordingly.

Note that if this achievement is a Tracked Persistent achievement, and the underlying PPV value still exceeds the configured threshold, marking such an achievement incomplete will have no effect. It will remain completed.

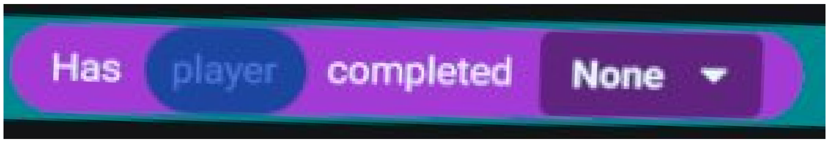
Operations

has player completed achievement



This operation returns a boolean value, and allows you to check whether an achievement has been previously completed for a particular player. This lets you decouple actions based on achievement completion from when they are actually completed.

When placed in a script, the full syntax of the code block is show below:



Parameters of the code block are as follows:

player

You must provide a reference to the player for whom you want to check achievement status. This player object can come from an existing valuable, or via a parameter to an event handler (like trigger entry or collision).

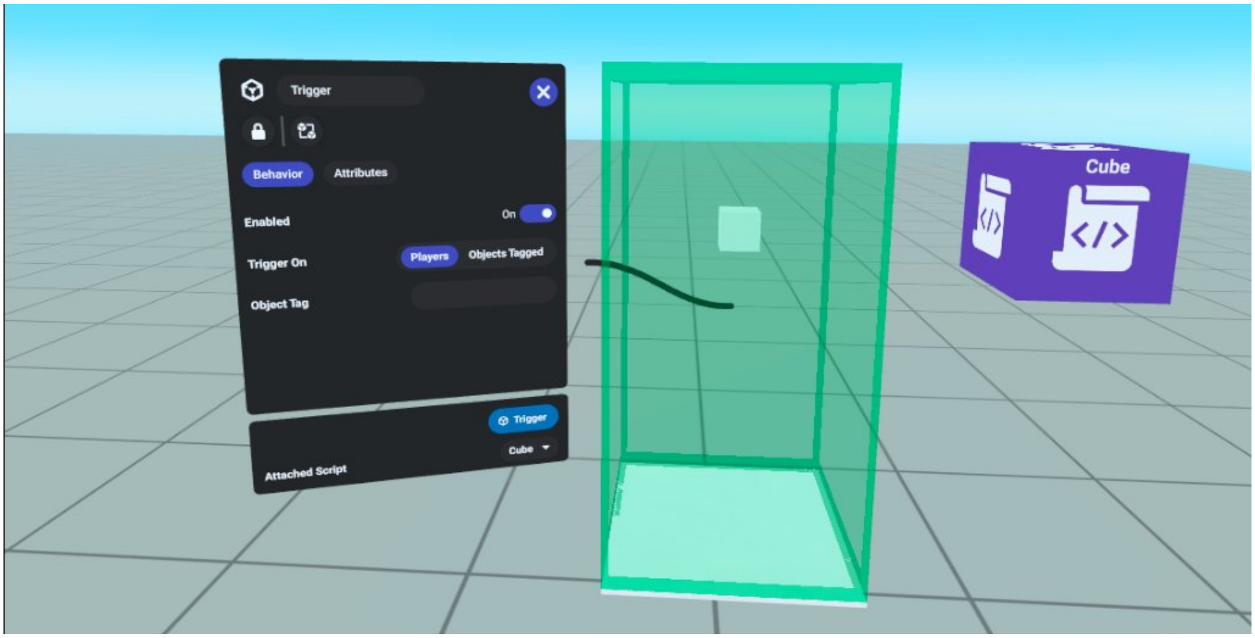
achievement

This is a drop down list of the Name values of all configured achievements. It is *not* the Script ID. Select the Name of the achievement whose state you want to check.

Examples

Simple Achievement

In the following example, there is a Player Trigger with a script attached that sets the ‘Cube Hello’ achievement complete when the player enters the trigger, and incomplete when the player leaves the trigger.



Trigger

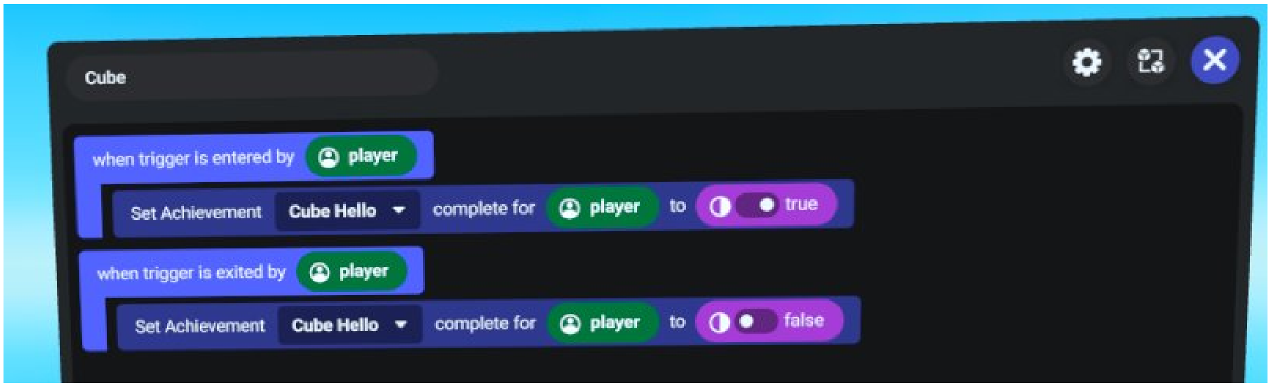
Behavior | Attributes

Enabled: On

Trigger On: Players | Objects Tagged

Object Tag

Attached Script: Trigger Cube



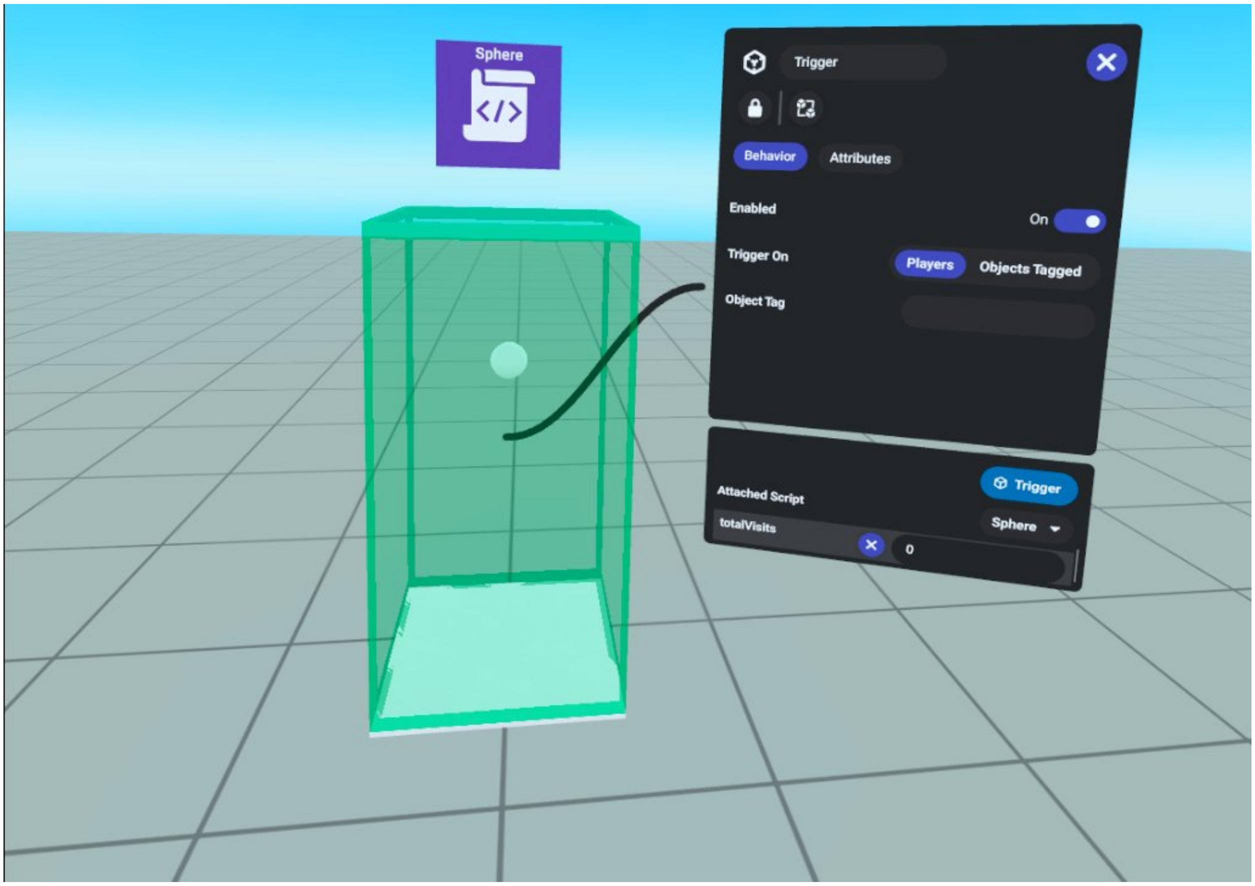
Cube

```
when trigger is entered by player
    Set Achievement Cube Hello complete for player to true
when trigger is exited by player
    Set Achievement Cube Hello complete for player to false
```

The achievement to mark complete or incomplete is referenced by Name, not ScriptID, via the drop down list.

Tracked Persistent Achievement

In the following example, a Player Persistent Variable (PPV) called ‘sphereVisits’ has been configured in the Systems->Player configuration screen. Each entry into the trigger increments the value of the ‘sphereVisits’ PPV.



Trigger

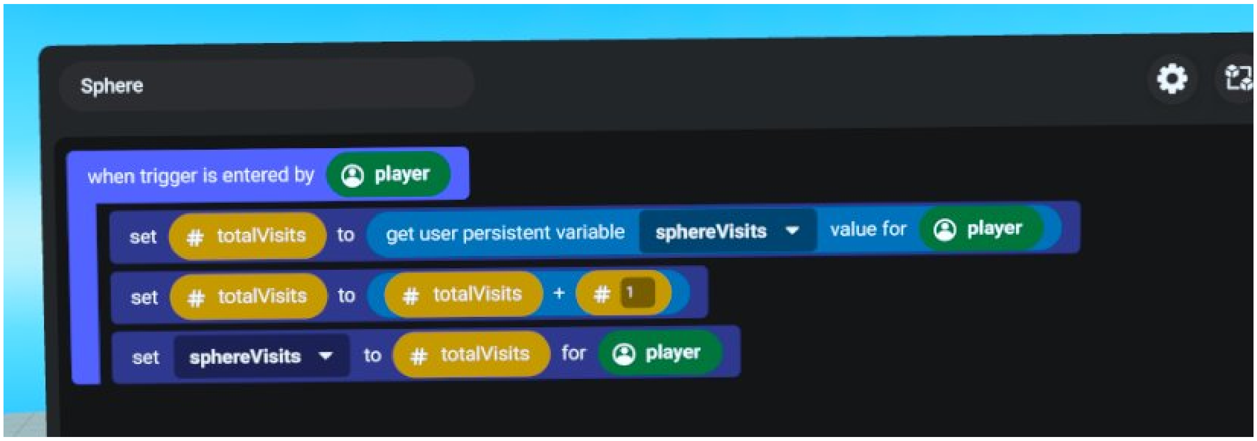
Enabled: On

Trigger On: Players | Objects Tagged

Object Tag

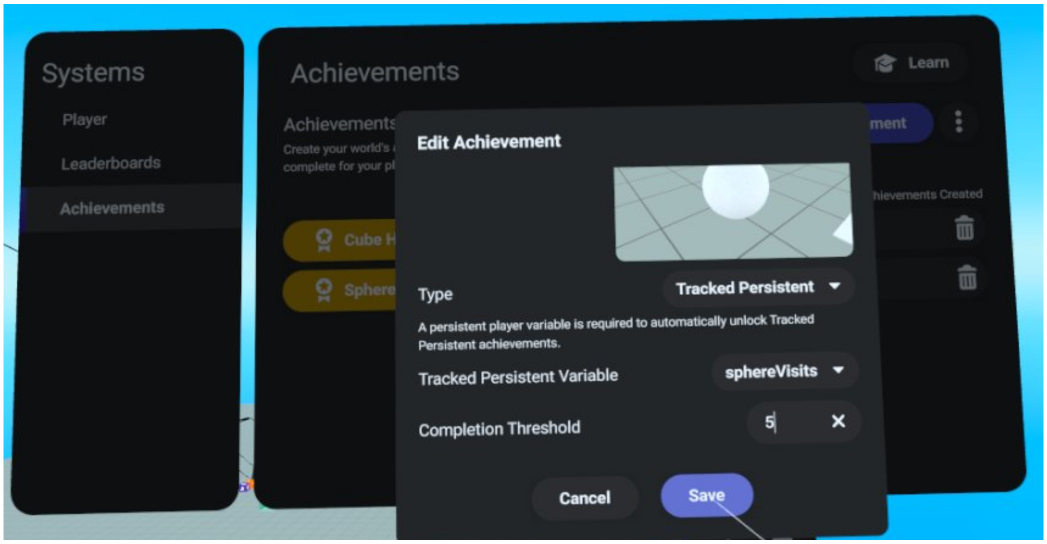
Attached Script: Sphere

totalVisits: 0



```
Sphere
when trigger is entered by player
    set totalVisits to get user persistent variable sphereVisits value for player
    set totalVisits to totalVisits + 1
    set sphereVisits to totalVisits for player
```

Note that there is no code in the script that directly completes the achievement. It only adjusts the PPV value.



Edit Achievement

Type: Tracked Persistent

A persistent player variable is required to automatically unlock Tracked Persistent achievements.

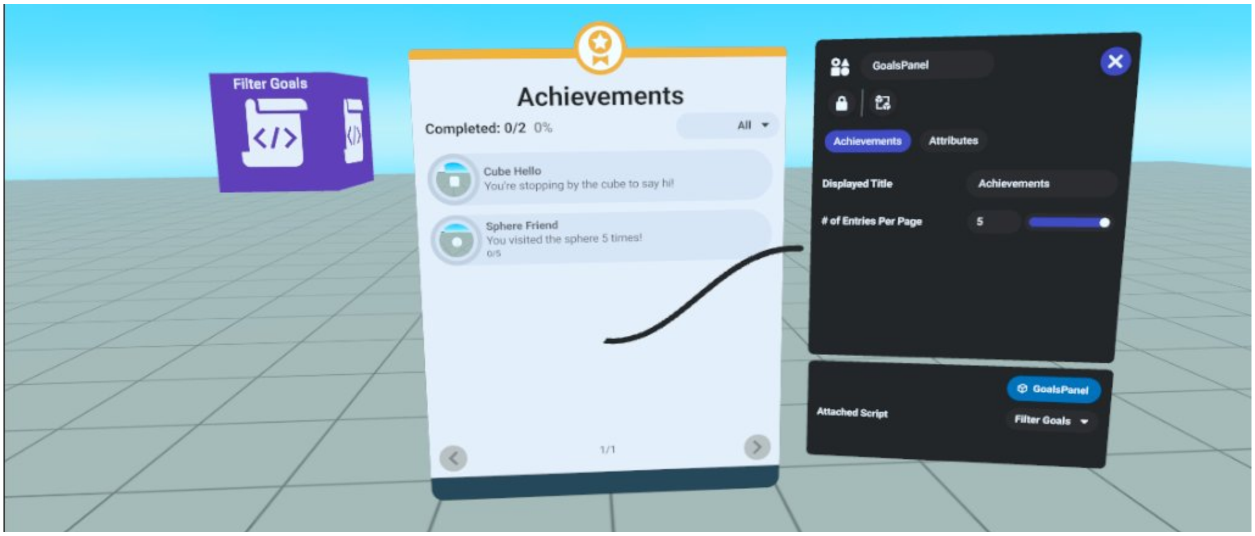
Tracked Persistent Variable: sphereVisits

Completion Threshold: 5

There is a Tracked Persistent Achievement configured to auto-complete when the ‘sphereVisits’ PPV reaches certain thresholds, in this case ‘5’.

Achievement Panel Filters

The following example shows how you can use a script to filter and reorder the items that appear on an achievements panel gizmo.



Achievements

Completed: 0/2 0% All

Cube Hello
You're stopping by the cube to say hi!

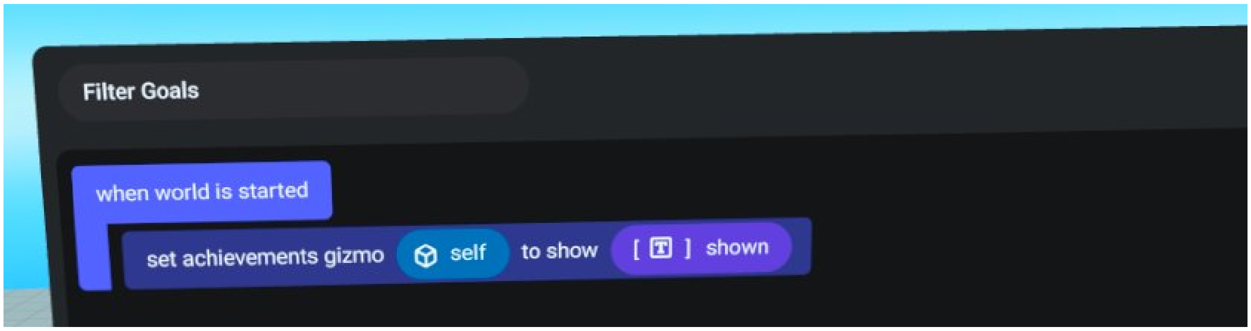
Sphere Friend
You visited the sphere 5 times!
0/5

1/1

Displayed Title: Achievements

# of Entries Per Page: 5

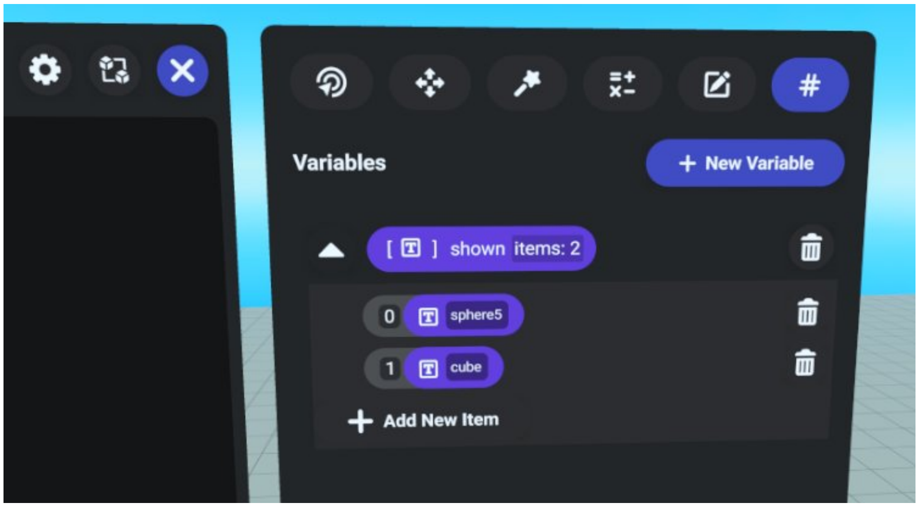
Attached Script: GoalsPanel, Filter Goals



Filter Goals

```
when world is started
    set achievements gizmo [self] to show [ [T] shown ]
```

You simply need to set a list of Script ID strings for the achievements (not the Names of the achievements) you want displayed, in the order you want them displayed, on the script gizmo.



[T] shown items: 2

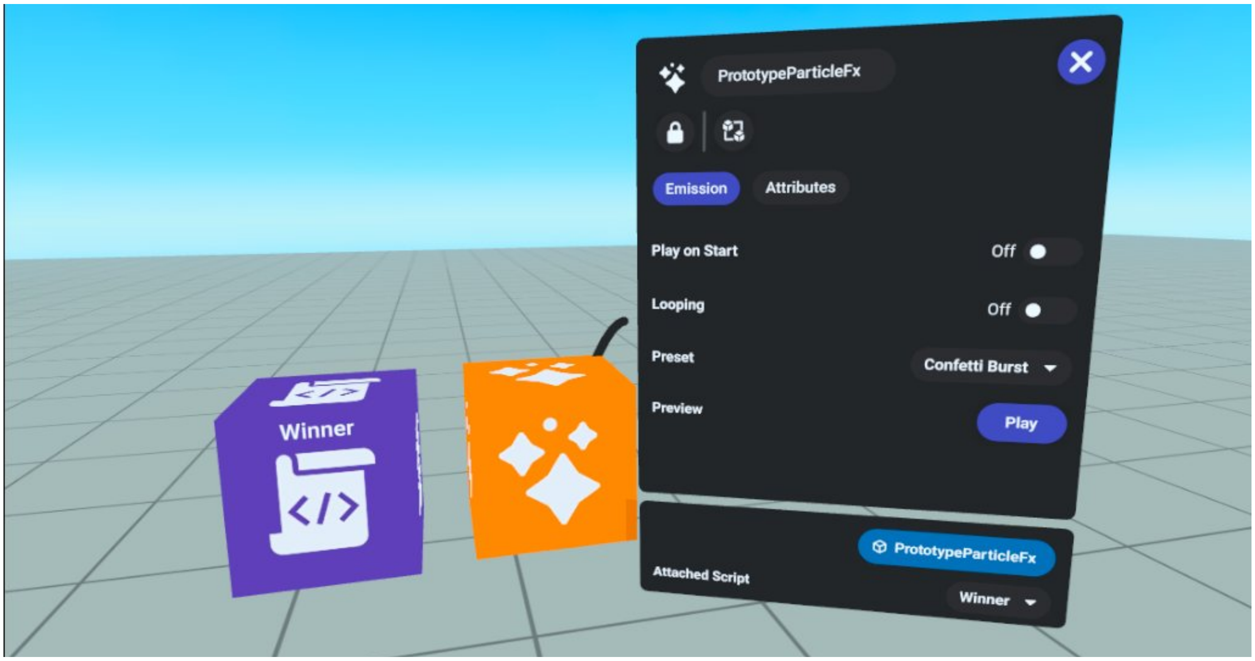
0 sphere5

1 cube

This is an ideal use case for the Variable panel inline configuration of default value entries for text lists. In this case, the Script IDs of the achievements we want displayed are ‘sphere5’ and ‘cube’.

Achievement Events

This example shows a particle effect with a script attached that monitors for the completion of achievements.



PrototypeParticleFx

Emission | Attributes

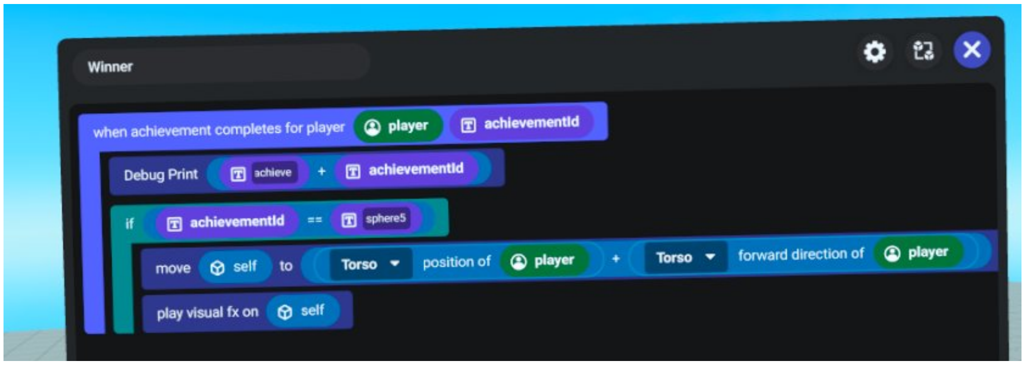
Play on Start: Off

Looping: Off

Preset: Confetti Burst

Preview: Play

Attached Script: PrototypeParticleFx — Winner



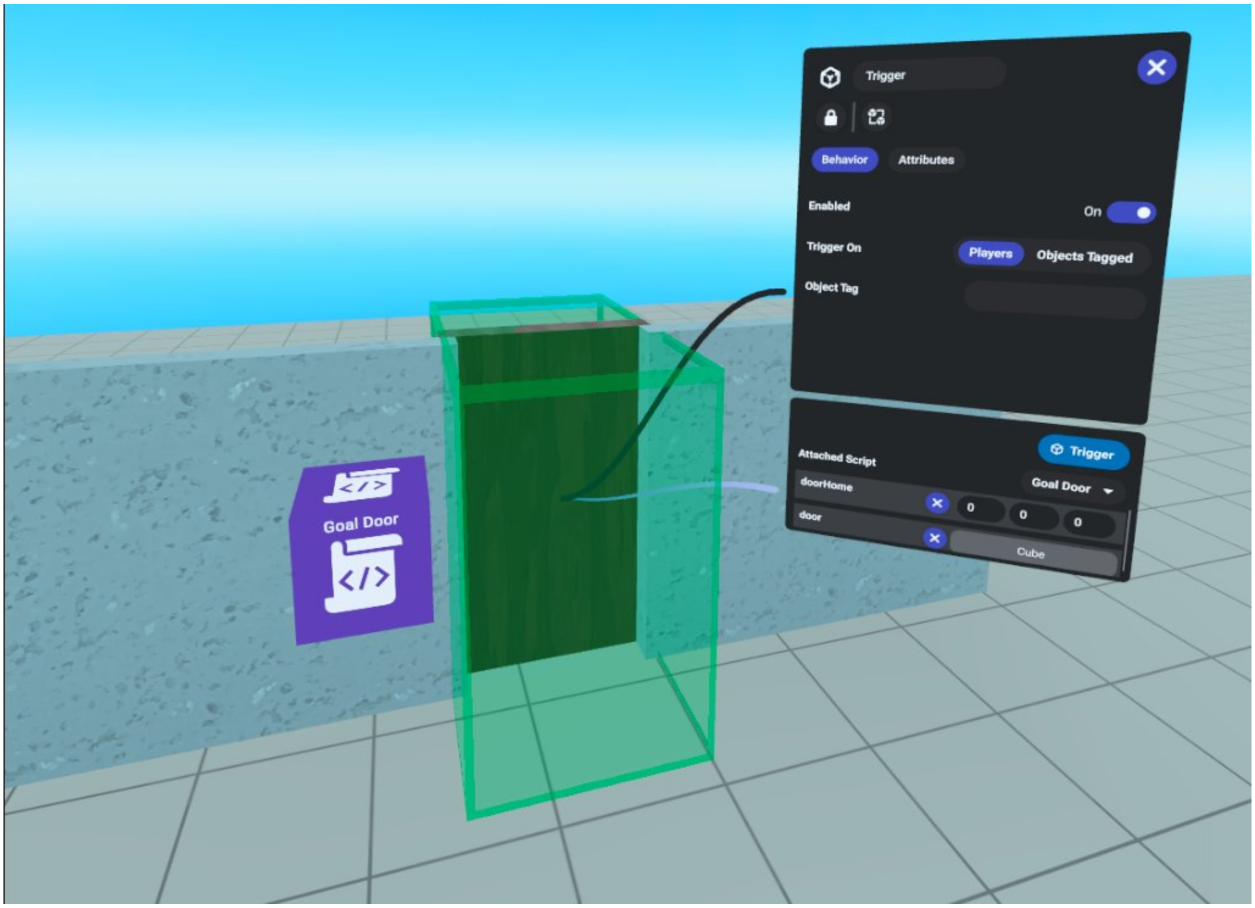
Winner

```
when achievement completes for player  player  achievementId
    Debug Print  achieve + achievementId
    if  achievementId == sphere5
        move  self  to  Torso position of  player  +  Torso forward direction of  player
        play visual fx on  self
```

Note that an event is received for every achievement completed by every player. If you want to detect a specific achievement, you need to compare the ‘achievementId’ with the Script ID (not the Name) of the achievement you care about. When found (in this case the ‘sphere5’ achievement), this script moves and plays the particle effect in front of the player completing the achievement.

Achievement Operations

In this example, we have a trigger in front of a door that only will open if the player has previously completed some achievement.



Trigger

Behavior | Attributes

Enabled: On

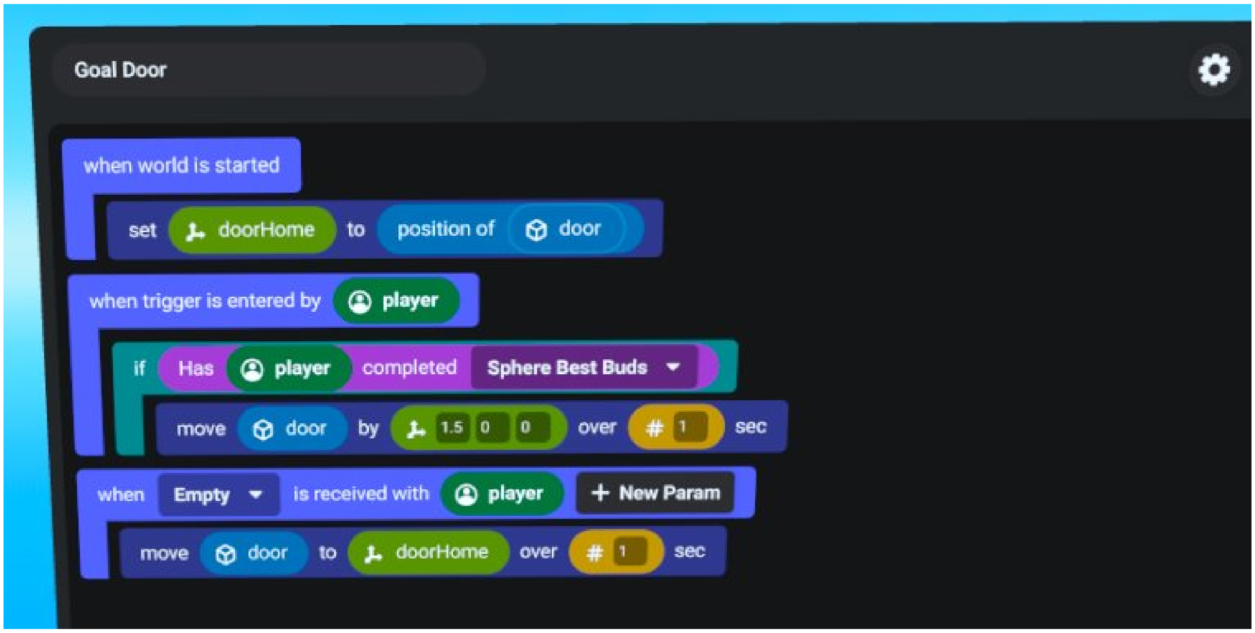
Trigger On: Players | Objects Tagged

Object Tag:

Attached Script: Trigger — Goal Door

doorHome: 0 0 0

door: Cube



Goal Door

```
when world is started
    set doorHome to position of door
when trigger is entered by player
    if Has player completed Sphere Best Buds
        move door by 1.5 0 0 over 1 sec
when Empty is received with player + New Param
    move door to doorHome over 1 sec
```

When the trigger is entered by the player, we check if the player has completed the Achievement (looked up by Name, not ScriptID) at some point in the past, and if so, slide the door off to the side. When all players have left the trigger, the door is returned to its closed position.

Design Advice

You are going to want to design achievements that are fun for players to complete. If you make your achievements too confusing, difficult, obscure, or uninteresting, the player will either not undertake the task of completing them, or will feel short-changed for spending time on something that in the end they perceive as ‘not worth it’. While some achievements can be happened upon accidentally by players (e.g.. you got an achievement for visiting the world 100 times, but you did so without trying to check something off a list; you just like this world!), many players are going to follow whatever direction you make visible in your Achievements Panel and strive to complete those goals.

Here’s a framework and an example for designing achievements that people will seek to earn.

Framework

Achievements are a key part of “player equity” where players have a sense of value, property, or investment in a world. Achievements give players direction on what to do and provide them with rewards upon their accomplishment. When designing an achievement consider the following questions:

1. **Clear motivation** - people have different motivations. Sometimes they want to be the best, or they want to find people with shared interests, or they want to own something unique, they want to just relax, and so many more possibilities. When building an achievement, think about who the player is and what motivates them.
2. **Aligned objective** - in order to earn the achievement you will need to do / experience / find something, at which point the achievement is unlocked. That action should be aligned with the motivation. The action “jump for the 10,000th time” would not match the objective “relax and unwind”.
3. **Manageable path** - a person should have some understanding of how to approach an achievement. It is often sufficient to simply tell them the objective “get to the bottom of the sea” and that gives them direction. But you don’t always have to make it so clear. “Find an egg” is an objective but also gives a person the direction to start looking. You can accomplish this with secret achievements too: if a world has 20 achievements and a player has 18 that gives them direction to find the remaining two. The harder it is to intuit the path forward, the less likely a person is to engage (unless the reward is enticing enough to overcome it).
4. **Meaningful reward** - an achievement requires time / effort / energy (the “cost”) which must be worth it (the “value”). If an achievement takes 100 hours to get, and upon completion a funny sound plays (and nothing else at all) people will likely be quite disappointed. If a reward isn’t enticing enough or the effort seems too high, people won’t even try. If people think it is worthwhile but then find out it was not then they will lose trust / interest with the creator and their content.

Example

Let’s look at an example. Imagine we have a rock climbing world. We are going to design an achievement for the serious competitors, people that are motivated to push themselves, work hard to be the best, and are hungry for a challenge. Let’s focus on the **internal motivations** of *being really good at something* and the **external motivation** of the status / fame for being the best.

We’ll imagine an achievement for climbing the rockwall in under 20s (assuming in our imaginary world that is insanely fast!). This **objective** is aligned with the motivations we chose because in order to accomplish this you must be highly skilled.

In addition to an Achievement, we could add a Leaderboard of “fastest times” to the world. This will help a person understand their rank. Has anyone gotten close to the 20s limit yet? Many people? Just one? Where are you in the list? This will give people a **path** (e.g. getting in the top 10, shaving their top time down just another 5s, etc).

To turn reaching the 20s speed into an Achievement, let’s make the **reward** be colorful trails that flow from your hands. This will make it so that you look really cool while climbing and people will say “how did you get those?!” which ties back to the motivation of being recognized as skilled. Other people will see you with them and that will give them purpose / direction to try and earn them too.

Secret achievements

Using Achievement Board Filtering code blocks, you could make achievements that are completely secret, and surprisingly this doesn’t go against the framework described above

Perhaps when a person puts a donut in a bank vault (with no clues to do so) they suddenly unlock a laser sword. That person didn’t have motivation or purpose but now that they have the laser sword, anyone who sees them with it will now have a goal / purpose to get it, too. So for everyone else the framework still applies.

Creating secret achievements means that the first people to find it rely on luck but everyone after will follow the same framework of motivation, objective, path, reward. Secret achievements allow for some mystery and intrigue but run the risk of being so obscure as to never be completed.

Example

Secret achievements are good for rewarding people who find ‘easter egg’ hidden spaces or objects in your world. You probably don’t want to advertise the existence of all these easter eggs on your Achievements board, but you would want to let the player know when they found one of them.

You could even have a non-secret achievement of “Found all the Easter Eggs” or “Found all 10 Easter Eggs” (which uses a Tracked Persistent achievement to measure progress towards finding them all). That would motivate people to know that secret achievements exist and seek them out, but not provide any hints for each *individual* easter egg achievement.

Current Limitations and Bugs (as of August 2022)

Some parts of the Achievements system work in unexpected ways. Some of these may be bugs that will be fixed over time, but others are built in characteristics of the system that may be non-intuitive to some world builders.

Editing/Resetting Tracked Persistent Achievements

Resetting Tracked Persistent achievements via the Achievements UI will not have any effect *if* the underlying PPV thresholds are still met. There is no notification that they have ‘auto completed’ again after the reset request. They just don’t appear to change at all. To truly reset achievements, you must *first* reset the PPV below the threshold *before* resetting the Tracked Persistent achievements via the UI.

World Editors can’t reset Tracked Persistent Achievements via UI

Since the PPV UI only allows World Owners to adjust anything having to do with PPVs, including resetting their values, World Editors are unable to reset their PPVs being tracked by Tracked Persistent achievements below the threshold value in build mode. Thus, they can never reset a Tracked Persistent achievement via the UI alone. Instead, they will need to write code executed during preview mode that sets their PPV to a new value when their avatar interacts with a trigger or grabbable object or something else that provides a player variable to an event handler. After resetting the PPV, they can then use the UI to reset their Tracked Persistent achievement.

Completed Tracked Persistent Achievements never auto-incomplete

If the PPV being tracked by a tracked persistent achievement drops below the configured threshold for that achievement, the achievement is NOT marked incomplete. It remains completed. In-progress Persistent Tracked achievements, however, do update progress as PPV is changed, and will roll back progress as the PPV decreases.

Tracked Persistent Achievements don’t always obey explicit completion code blocks

If you use the “set accomplishment for player” code blocks to explicitly set the completion state to true or false for a tracked persistent achievement, it may not ‘take’ on the first invocation. You may need to run the code block a second time for it to take effect.

Stuck Tracked Persistent Achievements

Some tracked persistent achievements seem to get ‘semi-stuck’ in that they will not pop up the completion dialog for the player or send events to script blocks when they get completed, but *will* update the leaderboard to show they are complete. Leaving the build session and coming back into a new session seems to fix this.

Achievement Board Filtering sometimes doesn’t work at first

If you are having trouble getting an achievement board to accept a list of strings for achievements to display (i.e. getting an “Unable to execute action setAchievementsList” in the scripting console), try exiting entirely out of the world, waiting a few minutes for the server to shut down the build instance, and then going back into build mode. It may not work initially upon return, either, but if you stop/start the world once, it should start working after that point.

Achievement Board Ordering not obeyed

The documentation says that the order of the achievement ‘script ids’ in the string list passed to “set achievement gizmo <gizmo> to show <list>” should dictate the order that the achievements are displayed in the gizmo, but it doesn’t seem to always work deterministically. Some achievements will be in order, while others will not.

Only the World Owner can make persistent changes to Achievements Settings

World Editors can *appear* to make changes to achievements, such as changing their Script ID, that appear to be saved (in that they continue to show the new values in the Achievements settings UI, even after world stop/start), but such changes will not be persisted in the saved world state when the build instance is shut down. Only changes made by the World Owner persist across resetting of the whole build instance.