

Areas

Areas are oblong-shaped collision zones which can trigger callbacks when the player or other objects enter them; they are also often used as position markers for things spawned in code such as sounds or particle effects.

HPL3 ships with several different area types - here's a quick summary of them and of some of their properties.

Trigger

The most commonly used type of area, used for all sorts of purposes.

Important Properties

BlocksLineOfSight

If true, this area acts as a blocker for AI seeing the player.

CC_Entities, CC_Funcs

Collision callbacks - for example, if you set CC_Entities to `player` and CC_Funcs to `TriggerHitArea` then `TriggerHitArea` will be called in your map script whenever the player enters or leaves the area.

PlayerLookAtCallback

Called when the player has the area on-screen. Various other properties in the same section modify exactly when that happens.

PlayerInteractCallback

Makes the area interactable, and triggers this callback when the player interacts with it. You can also set `CustomInteractIcon` etc.

AmbientLight

The player, by default, casts a small point light around him so that you can see things even when the room is very dark. This area type changes that ambient light to fit the current area.

Important Properties

Type

Indoors / Outdoors / None

CameraAnimation

Used as a camera viewpoint in a camera animation. See [Camera Animation](#).

Climb

This area becomes interactable; when you interact, you will climb up to a specified foot position. It's used for things like climbing into vent shafts or climbing up on to a nearby table or something; for higher climbs, you probably want a camera animation or a ladder instead.

Important Properties

FeetPosition

Name of an entity (usually a trigger area) specifying where the feet should land after the climb.

Crouch

Specify if the player should be automatically crouched after the climb (useful for vent shafts).

Crawl

The player will be crawling on their belly while in this area, and can't run or stand up. Useful for vent shafts.

Datamine

The player can interact with this area as if it were a datamine prop - see [Datamine](#)

DatamineAudioSource, DatamineAnimNode

Currently unused.

Description

Used with the [Description](#) module. Not used in SOMA.

Distortion

This area will apply a [Distortion Effect](#) to the player while they are in it.

DoorwayTrigger

This area is really useful for telling you when the player has walked into a particular part of the map. You set up a doorway with the Z-axis pointing in to the 'inside' and away from the 'outside'. Now when the player walks in through the door, the DoorwayCallback is called with an `alState` parameter of 1. When the player walks out through the door, the callback is called with an `alState` of -1.

You can also link sets of doorways, which is a great way to figure out if the player is inside an irregular space. If you specify the DoorwayGroup property, then you can use the function `Doorway_PlayerIsInGroup` to check if the player has crossed into the space delimited by these doorways. (Essentially, if you cross any of the doorways going 'inwards' so `alState==1`, then you are in the group; crossing any going 'outwards' so `alState==-1` means you are outside the group).

Important Properties

DoorwayEntity

This acts just like `CC_Entities` on an area - it just specifies which entity to check for collisions with. This is usually `player`.

DoorwayCallback

Called when the `DoorwayEntity` goes in or out of the doorway.

From:

<https://wiki.frictionalgames.com/> - **Frictional Game Wiki**

Permanent link:

<https://wiki.frictionalgames.com/hpl3/game/areas?rev=1442478915>

Last update: **2015/09/17 09:35**

