Chyron

VSAR 2.0.0

API Guide

Contents

INTRODUCTION

This document explains the API to remote control VSAR – Unreal Engine 5.

Help and Support

For contact information or our online helpdesk, please visit our support page at chyronhego.com/support/overview.

Disclaimer: Our products are subject to continual development and improvement. Therefore, while the information in this document was complete and accurate when it was written, additions or modifications to the products may cause changes to the technical and functional specifications. No rights can be derived from this document.

CONNECTION

Each VSAR server is connected to a Data Engine instance. PRIME VSAR is listening to bucket: “ue4” and key: “lua_in”

Note: “ue4” name is retained from ue4 due to backward compatibility

ACTOR CLASS COMMANDS

Actor(InActorID, index)

The constructor tyres to match the unique ActorID first, if it's unsuccessful it fallbacks to ActorName (label)

If the index is set to true or not set it will search for the first occurrence of the ActorName.

If an index is set to a number and there are multiple occurrences it will act like an index.

For example:

There are 4 actors named "Chair" in the level (ChairStuff, Chair_2, ChairStuff_16, ChairStuff)

Actor("Chair_2") will contain the Chair_2

Actor("Chair", true) will contain the first occurrence found of Chair

Actor("Chair", 2) will contain the third occurrence found of chair

Actor("Chair") will contain the first occurrence found of Chair

"Chair_2" has ID "StaticMeshActor_2"

Actor("StaticMeshActor_2") will contain the Chair_2

In the other classes/functions when there is a parameter named InActorID with another parameter Index it refers to this behavior.

GetName()

Returns the name of the actor (as a FString)

Example:

PrintLog(Actor("Cube1").GetName()) -- prints the name of the found actor

Move(x, y, z)

Move an actor by actor ID.

Example

Actor(“Cube1”).Move(960, 540, 0)

Rotate(pitch, yaw, roll)

Rotate an actor by actor ID

Example

Actor(“Cube1”).Rotate(360, 0, 0)

Scale(x, y, z)

Scale an actor by actor ID.

Example

Actor(“Cube1”).Scale(1, 10, 1)

Show( )

Show an actor by actor ID.

Example

Actor(“Cube1”).Show()

Hide( )

Hide an actor by actor ID.

Example

Actor(“Cube1”).Hide()

SetVisibility(bIsVisible)

Sets the visibility of the actor by ID.

Example

Actor(“Cube1”).SetVisibility(true)

CallBlueprintFunction(InFunctionName, Parameters ...)

Calls the actors blueprint function by name

Examples

Calling a function without parameter

Actor("TestLua").CallBlueprintFunction("FuncNoParam")

Calling a function with a string parameter

Actor("TestLua").CallBlueprintFunction("FuncStrParam", "Test")

Calling a function with a number parameter

Actor("TestLua").CallBlueprintFunction("FuncNumberParam", 42)

Calling a function with a vector parameter

Actor("TestLua").CallBlueprintFunction("FuncVectorParam", "(X=1.0 , Y=2.0 ,Z=3.0)")

\Calling a function with two parameters: LinearColor and bool

Actor("TestLua").CallBlueprintFunction("FuncTwoParams", "(R=1.0, G=0.0, B=0.5, A=1.0)", false)

SetVar(InVarName, InValue)

Set an actor variable by actor name and variable path/name.

Examples

Set a float value

Actor("TestLua", true).SetVar("FloatVar", 10.0)

Set a bool value

Actor("TestLua", true).SetVar("BoolVar", true)

Set X value in a Vector

Actor("TestLua", true).SetVar("VectorVar.X", 42)

We can also set a vector

Actor("TestLua", true).SetVar("VectorVar", ToFVector(10, 20, 30))

We can set a FLinearColor

Actor("TestLua", true).SetVar("LinearColorVar", ToFLinearColor(1, 0, 0, 1))

Or just the R value

Actor("TestLua", true).SetVar("LinearColorVar.G", 1.0)

GetVar(InVarName, InValue)

When using GetVar it returns a json string, containing the type, the name and the value of the var

Examples

Get the json string of the var “FloatVar”

local floatJson = Actor("TestLua", true).GetVar("FloatVar")

Decode the string

local floatDecoded = JSON:decode(FromFString(floatJson))

Print the value of FloatVar

PrintLog(floatDecoded.Name .. “=” .. floatDecoded.Value)

This also works for structs:

Get the json string of “VectorVar”

local vectorJson = Actor("TestLua", true).GetVar("VectorVar")

Decode the string

local vectorDecoded = JSON:decode(FromFString(vectorJson))

iterate and print each properties

for i,v in pairs(vectorDecoded.Properties) do

PrintLog(v.Name .. "=" .. v.Value)

end

COMPONENT COMMANDS

TypeOfComponent()

GetComponentByName(strNameOfComponentType)

Get reference to component in Actor defined by his name of type.

```
local actor1 = Actor("TestActor1");

PrintLog(actor1.GetName());

local component = actor1.GetComponentByName("MtCamio2DText");

PrintLog(component.GetName());
```

PropertyTools.CallFunction(component.ObjectPtr, strNameOfFunction)

Call function without parameter defined for component.

```
local actor1 = Actor("TestActor1");

PrintLog(actor1.GetName());

local component = actor1.GetComponentByName("MtCamio2DText");

PrintLog(component.GetName());

PropertyTools.CallFunction(component.ObjectPtr, "Play" );
```

PropertyTools.CallFunction(component.ObjectPtr, strNameOfFunction, strPar1)

Call function with one parameter defined for component.

```
local actor1 = Actor("TestActor1");

PrintLog(actor1.GetName());

local component = actor1.GetComponentByName("MtCamio2DText");

PrintLog(component.GetName());

PropertyTools.CallFunction(component.ObjectPtr, "SetXScale",  "1.0");
```

PropertyTools.CallFunction(component.ObjectPtr, "SetYScale", "1.0");

PropertyTools.CallFunction(component.ObjectPtr, strNameOfFunction, strPar1, strPar2)

Call function with two parameters defined for component.

LEVEL SEQUENCE CLASS COMMANDS

LevelSequence(InActorID)

Play( )

Play a sequence in a level by actor name.

Example

LevelSequence("MySequence").Play()

Pause( )

Pause a sequence in a level by actor name.

Example

LevelSequence("MySequence").Pause()

PlayReverse( )

Play a sequence reversed in a level by actor name.

Example

LevelSequence("MySequence").PlayReverse()

PlayToSeconds(seconds)

Play a sequence until the outpoint in a level by actor name and value.

Example

LevelSequence("MySequence").PlayToSeconds(5.0)

ScrubToSeconds(seconds)

Seek a sequence in a level by actor name and value in seconds.

Example

LevelSequence(“MySequence”).ScrubToSeconds(1.0)

PlayRate(playRate)

Set the play speed of the sequence. The play rate is 1.0 based. 1.0 is default speed, 2.0 is 2 times the speed, 0.5 half of the speed.

Example

LevelSequence(“MySequence”).PlayRate(1.0)

TimeRange(startTime, duration)

Set the sequence time from and time end values to play a smaller sequence by sequence name in seconds.

Example

LevelSequence(“Sequence”).TimeRange(1.0, 5.0)

AB SWITCH CLASS COMMANDS

MtABSwitch(InActorID)

There are general commands and per player commands

Parameters between “[ ]” are optional.

General Commands

SetEffectDuration (InDuration)

Example

MtABSwitch(“CH_ABSwitch_1”).SetEffectDuration(5.0)

SetEffect(InEffectName)

Example

MtABSwitch(“CH_ABSwitch_1”).SetEffect(“PushLeft”)

StartTransition([InName], [InDuration])

Example

MtABSwitch(“CH_ABSwitch_1”).StartTransition()

MtABSwitch(“CH_ABSwitch_1”).StartTransition(“WipeLeft”)

MtABSwitch(“CH_ABSwitch_1”).StartTransition(“Fade”, 10)

Player commands:

There are 2 players: Program and Preview

OpenFile(InFilePath)

Example

MtABSwitch("CH_ABSwitch_1").Program.OpenFile("D:/content/testclip.mpg");

MtABSwitch("CH_ABSwitch_1").Preview.OpenFile("D:/content/testimage.png");

OpenVidigoInputStream(InStreamID)

Example

MtABSwitch("CH_ABSwitch_1").Program.OpenVidigoInputStream("inputA");

MtABSwitch("CH_ABSwitch_1").Preview.OpenVidigoInputStream("inputB");

OpenUrl(InUrl)

Example

MtABSwitch("CH_ABSwitch_1").Program.OpenUrl("DaveShm://InputA");

MtABSwitch("CH_ABSwitch_1").Preview.OpenUrl("File://D:/content/testclip.mpg");

Play( )

Example

MtABSwitch("CH_ABSwitch_1").Program.Play();

MtABSwitch("CH_ABSwitch_1").Preview.Play();

Pause( )

Example

MtABSwitch("CH_ABSwitch_1").Program.Pause();

MtABSwitch("CH_ABSwitch_1").Preview.Pause();

Rewind( )

Example

MtABSwitch("CH_ABSwitch_1").Program.Rewind();

MtABSwitch("CH_ABSwitch_1").Preview.Rewind();

SetLooping(bIsLooping)

Example

MtABSwitch("CH_ABSwitch_1").Program.SetLooping(true);

MtABSwitch("CH_ABSwitch_1").Preview.SetLooping(false);

IsLooping( )

Example

local isProgramLooping = MtABSwitch("CH_ABSwitch_1").Program.IsLooping();

local isPreviewLooping = MtABSwitch("CH_ABSwitch_1").Preview.IsLooping();

IsPlaying( )

Example

local isProgramPlaying = MtABSwitch("CH_ABSwitch_1").Program.IsPlaying("");

local isPreviewPlaying = MtABSwitch("CH_ABSwitch_1").Preview.IsPlaying("");

SetRate(InRate)

Example

MtABSwitch(“CH_ABSwitch_1”).Program.SetRate(2.0);

MtABSwitch(“CH_ABSwitch_1”).Preview.SetRate(0.5);

CESIUM CAMERA CLASS COMMANDS

CesiumCamera(InActorID, index)

AddActorToForeground(InActorName, index)

Adds the actor to the cameras key channel

Example

CesiumCamera("CesiumCamera", true).AddActorToForeground("Chair", true)

RemoveActorFromForeground(InActorName, index)

Removes the actor from the cameras key channel

Example

CesiumCamera("CesiumCamera", true).RemoveActorFromForeground("Chair", true)

AddActorToMatte(InActorName, index)

Adds the actor to che key channel but hiddes it from the fill channel

Example

CesiumCamera("CesiumCamera", true).AddActorToMatte("Chair", true)

RemoveActorFromMatte(InActorName, index)

Removes the actor from the key channel and put's it again in the fill channel

Example

CesiumCamera("CesiumCamera", true).RemoveActorFromMatte("Chair", true)

DATA ENGINE COMMANDS

DataEngine.WriteToKey(Bucket, Key, Value)

Write the value to the Bucket/Key of the engines currently connected Data Engine

Example

DataEngine.WriteToKey("TestBucket", "TestKey", ToFString("TestString !"))

LEVELS COMMANDS

Only works for sublevels while in PIE

LevelShow(SubLevelName)

Shows the sublevel (needs to be loaded)

Example

LevelShow("MySubLevel")

LevelHide(SubLevelName)

Hides the sublevel (needs to be loaded)

Example

LevelHide("MySubLevel")

OpenLevel(levelName)

Permanently opens a level by it's name

Example

OpenLevel("NewLevel")

OpenLevelFast(levelName)

Temporarily opens a level by it's name, Works only in PIE, Can be Faster then OpenLevel

Example

OpenLevelFast("NewLevel")

EDITOR COMMANDS

StartPIE()

Starts the Play in editor (PIE)

StopPIE()

Stops the Play in editor (PIE)

OpenProject (projectPath)

Opens project if the provided path is valid. VSAR needs to be already running.

Example

OpenProject("C:\\VSAR\\Temp\\MyProject5\\MyProject5.uproject")

LOGGING COMMANDS

PrintLog(Str)

Prints the string as a “Log” verbosity (grey)

Example

PrintLog(42)

PrintLog(Actor(“Chair”, true))

PrintLog_Warning(Str)

Prints the string as a “Warning” verbosity (yellow)

Example

PrintLog_Warning(42)

PrintLog_Warning(Actor(“Chair”, true))

PrintLog_Error(Str)

Prints the string as a “Error” verbosity (red)

Example

PrintLog_Error(42)

PrintLog_Error(Actor(“Chair”, true))

CONVERSION COMMANDS

ToFString(str)

Converts a string to FString

ToFVector(x, y, z)

Converts x, y, z to FVector

ToFVector2D(x, y)

Converts x, y to FVector2D

ToFRotator(pitch, yaw, roll)

Converts pitch, yaw, roll to FRotator

ToFLinearColor(r, g, b, a)

Converts r, g, b, a to FLinearColor

FromFString(str)

Converts back FString to Lua

FromFVector(Vec)

Converts back FVector to Lua

FromFRotator(Rot)

Converts back FRotator to Lua