using UnrealBuildTool;

public class Atestat1 : ModuleRules

{

public Atestat1(ReadOnlyTargetRules Target) : base(Target)

{

PCHUsage = PCHUsageMode.UseExplicitOrSharedPCHs;

PublicDependencyModuleNames.AddRange(new string[] { "Core", "CoreUObject", "Engine", "InputCore" });

PrivateDependencyModuleNames.AddRange(new string[] { });

// Uncomment if you are using Slate UI

// PrivateDependencyModuleNames.AddRange(new string[] { "Slate", "SlateCore" });

// Uncomment if you are using online features

// PrivateDependencyModuleNames.Add("OnlineSubsystem");

// To include OnlineSubsystemSteam, add it to the plugins section in your uproject file with the Enabled attribute set to true

}

}

#include "Atestat1.h"

#include "Modules/ModuleManager.h"

IMPLEMENT\_PRIMARY\_GAME\_MODULE( FDefaultGameModuleImpl, Atestat1, "Atestat1" );

#pragma once

#include "CoreMinimal.h"

#include "Atestat1GameModeBase.h"

#pragma once

#include "CoreMinimal.h"

#include "GameFramework/GameModeBase.h"

#include "Atestat1GameModeBase.generated.h"

/\*\*

\*

\*/

UCLASS()

class ATESTAT1\_API AAtestat1GameModeBase : public AGameModeBase

{

GENERATED\_BODY()

};

using UnrealBuildTool;

using System.Collections.Generic;

public class Atestat1Target : TargetRules

{

public Atestat1Target(TargetInfo Target) : base(Target)

{

Type = TargetType.Game;

ExtraModuleNames.AddRange( new string[] { "Atestat1" } );

}

}

using UnrealBuildTool;

using System.Collections.Generic;

public class Atestat1EditorTarget : TargetRules

{

public Atestat1EditorTarget(TargetInfo Target) : base(Target)

{

Type = TargetType.Editor;

ExtraModuleNames.AddRange( new string[] { "Atestat1" } );

}

}

{

"FileVersion": 3,

"EngineAssociation": "4.22",

"Category": "",

"Description": "",

"Modules": [

{

"Name": "Atestat1",

"Type": "Runtime",

"LoadingPhase": "Default"

}

]

}<?xml version="1.0" encoding="utf-8"?>

<AutoVisualizer xmlns="http://schemas.microsoft.com/vstudio/debugger/natvis/2010">

<!-- Epic Games, Inc. UE4 Visualizers -->

<!-- Copy this into c:\Users\<Your user folder>\My Documents\Visual Studio 2012\Visualizers -->

<!-- FString visualizer -->

<Type Name="FString">

<DisplayString Condition="Data.ArrayNum == 0">Empty</DisplayString>

<DisplayString Condition="Data.ArrayNum &lt; 0">Invalid</DisplayString>

<DisplayString Condition="Data.ArrayMax &lt; Data.ArrayNum">Invalid</DisplayString>

<DisplayString Condition="Data.ArrayMax &gt;= Data.ArrayNum">{Data.AllocatorInstance.Data,su}</DisplayString>

<StringView Condition="Data.ArrayMax &gt;= Data.ArrayNum">Data.AllocatorInstance.Data,su</StringView>

</Type>

<!-- FText visualizer -->

<Type Name="FStringTableEntry">

<DisplayString>{\*DisplayString.Object}</DisplayString>

</Type>

<Type Name="FTextHistory\_StringTableEntry::FStringTableReferenceData">

<DisplayString Condition="StringTableEntry.Object != 0 &amp;&amp; StringTableEntry.WeakReferenceCount.ReferenceController->SharedReferenceCount &gt; 0">{\*StringTableEntry.Object}</DisplayString>

<DisplayString Condition="StringTableEntry.Object == 0 || StringTableEntry.WeakReferenceCount.ReferenceController->SharedReferenceCount == 0">&lt;MISSING STRING TABLE ENTRY&gt;</DisplayString>

</Type>

<Type Name="FTextHistory\_StringTableEntry">

<DisplayString>{\*StringTableReferenceData.Object}</DisplayString>

</Type>

<Type Name="TLocalizedTextData&lt;\*&gt;">

<DisplayString>{\*LocalizedString.Object}</DisplayString>

</Type>

<Type Name="TGeneratedTextData&lt;\*&gt;">

<DisplayString Condition="LocalizedString.Object">{\*LocalizedString.Object}</DisplayString>

<DisplayString Condition="!LocalizedString.Object">{DisplayString}</DisplayString>

</Type>

<Type Name="TIndirectTextData&lt;\*&gt;">

<DisplayString>{History}</DisplayString>

</Type>

<Type Name="FText">

<DisplayString>{\*TextData.Object}</DisplayString>

</Type>

<!-- FName visualizer -->

<Type Name="FName">

<DisplayString Condition="ComparisonIndex &gt;= 4194304">Invalid</DisplayString>

<DisplayString Condition="ComparisonIndex &lt; 0">Invalid</DisplayString>

<!-- BEGIN: WideName support -->

<DisplayString Condition="ComparisonIndex &lt; 4194304 &amp;&amp; Number &gt; 0 &amp;&amp; (((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[ComparisonIndex / 16384][ComparisonIndex % 16384]))->Index &amp; 1) == 1">{((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[ComparisonIndex / 16384][ComparisonIndex % 16384]))->WideName}\_{Number-1}</DisplayString>

<DisplayString Condition="ComparisonIndex &lt; 4194304 &amp;&amp; (((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[ComparisonIndex / 16384][ComparisonIndex % 16384]))->Index &amp; 1) == 1">{((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[ComparisonIndex / 16384][ComparisonIndex % 16384]))->WideName}</DisplayString>

<!-- END: WideName support -->

<DisplayString Condition="ComparisonIndex &lt; 4194304 &amp;&amp; Number &gt; 0">{((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[ComparisonIndex / 16384][ComparisonIndex % 16384]))->AnsiName}\_{Number-1}</DisplayString>

<DisplayString Condition="ComparisonIndex &lt; 4194304">{((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[ComparisonIndex / 16384][ComparisonIndex % 16384]))->AnsiName}</DisplayString>

<!-- BEGIN: WideName support -->

<StringView Condition="ComparisonIndex &lt; 4194304 &amp;&amp; (((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[ComparisonIndex / 16384][ComparisonIndex % 16384]))->Index &amp; 1) == 1">((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[ComparisonIndex / 16384][ComparisonIndex % 16384]))->WideName</StringView>

<!-- END: WideName support -->

<StringView Condition="ComparisonIndex &lt; 4194304">((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[ComparisonIndex / 16384][ComparisonIndex % 16384]))->AnsiName</StringView>

</Type>

<Type Name="FName">

<DisplayString Condition="DisplayIndex &gt;= 4194304">Invalid</DisplayString>

<DisplayString Condition="DisplayIndex &lt; 0">Invalid</DisplayString>

<!-- BEGIN: WideName support -->

<DisplayString Condition="DisplayIndex &lt; 4194304 &amp;&amp; Number &gt; 0 &amp;&amp; (((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[DisplayIndex / 16384][DisplayIndex % 16384]))->Index &amp; 1) == 1">{((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[DisplayIndex / 16384][DisplayIndex % 16384]))->WideName}\_{Number-1}</DisplayString>

<DisplayString Condition="DisplayIndex &lt; 4194304 &amp;&amp; (((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[DisplayIndex / 16384][DisplayIndex % 16384]))->Index &amp; 1) == 1">{((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[DisplayIndex / 16384][DisplayIndex % 16384]))->WideName}</DisplayString>

<!-- END: WideName support -->

<DisplayString Condition="DisplayIndex &lt; 4194304 &amp;&amp; Number &gt; 0">{((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[DisplayIndex / 16384][DisplayIndex % 16384]))->AnsiName}\_{Number-1}</DisplayString>

<DisplayString Condition="DisplayIndex &lt; 4194304">{((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[DisplayIndex / 16384][DisplayIndex % 16384]))->AnsiName}</DisplayString>

<!-- BEGIN: WideName support -->

<StringView Condition="DisplayIndex &lt; 4194304 &amp;&amp; (((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[DisplayIndex / 16384][DisplayIndex % 16384]))->Index &amp; 1) == 1">((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[DisplayIndex / 16384][DisplayIndex % 16384]))->WideName</StringView>

<!-- END: WideName support -->

<StringView Condition="DisplayIndex &lt; 4194304">((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[DisplayIndex / 16384][DisplayIndex % 16384]))->AnsiName</StringView>

</Type>

<Type Name="FMinimalName">

<DisplayString Condition="Index &gt;= 4194304">Invalid</DisplayString>

<DisplayString Condition="Index &lt; 0">Invalid</DisplayString>

<DisplayString Condition="Index &lt; 4194304 &amp;&amp; Number &gt; 0">{((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[Index / 16384][Index % 16384]))->AnsiName}\_{Number-1}</DisplayString>

<DisplayString Condition="Index &lt; 4194304">{((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[Index / 16384][Index % 16384]))->AnsiName}</DisplayString>

<StringView Condition="Index &lt; 4194304">((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[Index / 16384][Index % 16384]))->AnsiName</StringView>

</Type>

<!-- FStatMessage visualizer @see Stats2.h -->

<!--

IsCycle = 0x04,

/\*\* if true, then this message contains a memory stat. \*/

IsMemory = 0x08,

/\*\* if true, then this is actually two uint32s, the cycle count and the call count, see FromPackedCallCountDuration\_Duration. \*/

IsPackedCCAndDuration = 0x10,

/\*\* if true, then this stat is cleared every frame. \*/

ShouldClearEveryFrame = 0x20,

-->

<Type Name="FStatMessage">

<!--ST\_None = 1 -->

<DisplayString Condition="((NameAndInfo.NameAndInfo.Number >> 9)&amp;7) == 1" >{{NoneType NameAndInfo={NameAndInfo.NameAndInfo}}}</DisplayString>

<!--ST\_int64 = 2 && !IsPackedCCAndDuration && !IsCycle -->

<DisplayString Condition="((NameAndInfo.NameAndInfo.Number >> 9)&amp;7) == 2 &amp;&amp; ((NameAndInfo.NameAndInfo.Number >> 9+3+4)&amp;0x10) != 0x10 &amp;&amp; ((NameAndInfo.NameAndInfo.Number >> 9+3+4)&amp;0x10) != 0x10" >

{{Int64={DebugStatData.Cycles} NameAndInfo={NameAndInfo.NameAndInfo}}}

</DisplayString>

<!--ST\_int64 = 2 && !IsPackedCCAndDuration && IsCycle -->

<DisplayString Condition="((NameAndInfo.NameAndInfo.Number >> 9)&amp;7) == 2 &amp;&amp; ((NameAndInfo.NameAndInfo.Number >> 9+3+4)&amp;0x10) != 0x10 &amp;&amp; ((NameAndInfo.NameAndInfo.Number >> 9+3+4)&amp;0x10) == 0x10" >

{{Cycles={DebugStatData.Cycles} NameAndInfo={NameAndInfo.NameAndInfo}}}

</DisplayString>

<!--ST\_int64 = 2 && IsPackedCCAndDuration && IsCycle -->

<DisplayString Condition="((NameAndInfo.NameAndInfo.Number >> 9)&amp;7) == 2 &amp;&amp; ((NameAndInfo.NameAndInfo.Number >> 9+3+4)&amp;0x10) == 0x10 &amp;&amp; ((NameAndInfo.NameAndInfo.Number >> 9+3+4)&amp;0x04) == 0x04" >

{{Count={DebugStatData.CCAndDuration[0]},Cycles={DebugStatData.CCAndDuration[1]} NameAndInfo={NameAndInfo.NameAndInfo}}}

</DisplayString>

<!--ST\_double = 3 -->

<DisplayString Condition="((NameAndInfo.NameAndInfo.Number >> 9)&amp;7) == 3" >{{Float={DebugStatData.Float} NameAndInfo={NameAndInfo.NameAndInfo}}}</DisplayString>

<!--ST\_FName = 4 -->

<DisplayString Condition="((NameAndInfo.NameAndInfo.Number >> 9)&amp;7) == 4" >{{Name={((FNameEntry\*)(((FNameEntry\*\*\*)GFNameTableForDebuggerVisualizers\_MT)[DebugStatData.Cycles / 16384][DebugStatData.Cycles % 16384]))->AnsiName} NameAndInfo={NameAndInfo.NameAndInfo}}}</DisplayString>

<!--ST\_Ptr = 5 -->

<DisplayString Condition="((NameAndInfo.NameAndInfo.Number >> 9)&amp;7) == 5" >{{Ptr={DebugStatData.Ptr} NameAndInfo={NameAndInfo.NameAndInfo}}}</DisplayString>

</Type>

<!-- FAllocationInfo -->

<!--

uint64 OldPtr;

uint64 Ptr;

int64 Size;

FName EncodedCallstack;

uint32 SequenceTag;

EMemoryOperation Op; Alloc=1, Free=2, Realloc=3

bool bHasBrokenCallstack;

-->

<Type Name="FAllocationInfo">

<!-- Alloc -->

<DisplayString Condition="Op == 1" >

{{A SeqTag={SequenceTag} Ptr={Ptr} Size={Size} Callstack={EncodedCallstack} bHasBrokenCallstack={bHasBrokenCallstack}}}

</DisplayString>

<!-- Free -->

<DisplayString Condition="Op == 2" >

{{F SeqTag={SequenceTag} Ptr={Ptr} bHasBrokenCallstack={bHasBrokenCallstack}}}

</DisplayString>

<!-- Realloc -->

<DisplayString Condition="Op == 3" >

{{R SeqTag={SequenceTag} OldPtr={OldPtr} Ptr={Ptr} NewSize={Size} Callstack={EncodedCallstack} bHasBrokenCallstack={bHasBrokenCallstack}}}

</DisplayString>

</Type>

<Type Name="FThreadSafeCounter">

<DisplayString>{Counter}</DisplayString>

</Type>

<Type Name="FThreadSafeBool">

<DisplayString Condition="Counter==0">False</DisplayString>

<DisplayString Condition="Counter==1">True</DisplayString>

</Type>

<!-- FTimespan visualizer -->

<Type Name="FTimespan">

<DisplayString>Ticks = {Ticks}</DisplayString>

<Expand>

<Item Name="Total Milliseconds">Ticks / ETimespan::TicksPerMillisecond</Item>

<Item Name="Total Seconds">Ticks / ETimespan::TicksPerSecond</Item>

<Item Name="Total Minutes">Ticks / ETimespan::TicksPerMinute</Item>

<Item Name="Total Hours">Ticks / ETimespan::TicksPerHour</Item>

<Item Name="Total Days">Ticks / ETimespan::TicksPerDay</Item>

</Expand>

</Type>

<Type Name="FVector\_NetQuantize">

<DisplayString>{{X={X} Y={Y} Z={Z}}</DisplayString>

</Type>

<Type Name="FVector\_NetQuantize10">

<DisplayString>{{X={X} Y={Y} Z={Z}}</DisplayString>

</Type>

<Type Name="FVector\_NetQuantize100">

<DisplayString>{{X={X} Y={Y} Z={Z}}</DisplayString>

</Type>

<Type Name="FVector\_NetQuantizeNormal">

<DisplayString>{{X={X} Y={Y} Z={Z}}</DisplayString>

</Type>

<!-- TEnumAsByte visualizer -->

<Type Name="TEnumAsByte&lt;\*&gt;">

<DisplayString>{($T1)Value}</DisplayString>

</Type>

<!-- UObjectBase visualizer -->

<Type Name="UObjectBase">

<DisplayString>(Name={NamePrivate})</DisplayString>

</Type>

<!-- TArray<\*,TFixedAllocator<\*> > visualizer -->

<Type Name="TArray&lt;\*,TFixedAllocator&lt;\*&gt;&gt;">

<DisplayString Condition="ArrayNum == 0">Empty</DisplayString>

<DisplayString Condition="ArrayNum &lt; 0">Invalid</DisplayString>

<DisplayString Condition="ArrayMax &lt; ArrayNum">Invalid</DisplayString>

<DisplayString Condition="ArrayMax &gt;= ArrayNum">Num={ArrayNum}</DisplayString>

<Expand>

<ArrayItems Condition="ArrayNum &lt;= ArrayMax">

<Size>ArrayNum</Size>

<ValuePointer>(TArray&lt;$T1,TFixedAllocator&lt;$T2&gt; &gt;::ElementType\*)AllocatorInstance.InlineData</ValuePointer>

</ArrayItems>

</Expand>

</Type>

<!-- TArray<\*,TInlineAllocator<\*,\*> > visualizer -->

<Type Name="TArray&lt;\*,TInlineAllocator&lt;\*,\*&gt;&gt;">

<DisplayString Condition="ArrayNum == 0">Empty</DisplayString>

<DisplayString Condition="ArrayNum &lt; 0">Invalid</DisplayString>

<DisplayString Condition="ArrayMax &lt; ArrayNum">Invalid</DisplayString>

<DisplayString Condition="ArrayMax &gt;= ArrayNum">Num={ArrayNum}</DisplayString>

<Expand>

<ArrayItems Condition="ArrayNum &lt;= ArrayMax">

<Size>ArrayNum</Size>

<ValuePointer Condition="AllocatorInstance.SecondaryData.Data == 0">(TArray&lt;$T1,TInlineAllocator&lt;$T2,$T3&gt; &gt;::ElementType\*)AllocatorInstance.InlineData</ValuePointer>

<ValuePointer Condition="AllocatorInstance.SecondaryData.Data != 0">(TArray&lt;$T1,TInlineAllocator&lt;$T2,$T3&gt; &gt;::ElementType\*)AllocatorInstance.SecondaryData.Data</ValuePointer>

</ArrayItems>

</Expand>

</Type>

<!-- TArray visualizer -->

<Type Name="TArray&lt;\*,\*&gt;">

<DisplayString Condition="ArrayNum == 0">Empty</DisplayString>

<DisplayString Condition="ArrayNum &lt; 0">Invalid</DisplayString>

<DisplayString Condition="ArrayMax &lt; ArrayNum">Invalid</DisplayString>

<DisplayString Condition="ArrayMax &gt;= ArrayNum">Num={ArrayNum}</DisplayString>

<Expand>

<ArrayItems Condition="ArrayNum &lt;= ArrayMax">

<Size>ArrayNum</Size>

<ValuePointer>(TArray&lt;$T1,$T2&gt;::ElementType\*)AllocatorInstance.Data</ValuePointer>

</ArrayItems>

</Expand>

</Type>

<!-- TArrayView visualizer -->

<Type Name="TArrayView&lt;\*&gt;">

<DisplayString Condition="ArrayNum == 0">Empty</DisplayString>

<DisplayString Condition="ArrayNum &lt; 0">Invalid</DisplayString>

<DisplayString Condition="ArrayNum &gt; 0">Num={ArrayNum}</DisplayString>

<Expand>

<ArrayItems Condition="ArrayNum &gt; 0">

<Size>ArrayNum</Size>

<ValuePointer>(TArrayView&lt;$T1&gt;::ElementType\*)DataPtr</ValuePointer>

</ArrayItems>

</Expand>

</Type>

<!-- TIndirectArray visualizer -->

<Type Name="TIndirectArray&lt;\*,\*&gt;">

<DisplayString Condition="Array.ArrayNum == 0">Empty</DisplayString>

<DisplayString Condition="Array.ArrayNum &lt; 0">Invalid</DisplayString>

<DisplayString Condition="Array.ArrayMax &lt; Array.ArrayNum">Invalid</DisplayString>

<DisplayString Condition="Array.ArrayMax &gt;= Array.ArrayNum">Num={Array.ArrayNum}</DisplayString>

<Expand>

<IndexListItems Condition="Array.ArrayNum &lt;= Array.ArrayMax">

<Size>Array.ArrayNum</Size>

<ValueNode>\*((TIndirectArray&lt;$T1,$T2&gt;::ElementType\*\*)Array.AllocatorInstance.Data)[$i]</ValueNode>

</IndexListItems>

</Expand>

</Type>

<!-- TChunkedArray visualizer -->

<Type Name="TChunkedArray&lt;\*,\*&gt;">

<DisplayString Condition="NumElements == 0">Empty</DisplayString>

<DisplayString Condition="NumElements &lt; 0">Invalid</DisplayString>

<DisplayString Condition="NumElements &gt; 0">NumElements={NumElements}, NumChunks={Chunks.Array.ArrayNum}, {NumElementsPerChunk}</DisplayString>

<Expand>

<IndexListItems Condition="NumElements &gt; 0">

<Size>NumElements</Size>

<ValueNode>

\*(

\*(

(TChunkedArray&lt;$T1,$T2&gt;::ElementType\*\*)Chunks.Array.AllocatorInstance.Data + ($i / NumElementsPerChunk)

) + ($i % NumElementsPerChunk)

)

</ValueNode>

</IndexListItems>

</Expand>

</Type>

<!-- TSparseArray visualizer -->

<Type Name="TSparseArray&lt;\*,\*&gt;">

<DisplayString Condition="(Data.ArrayNum - NumFreeIndices) &lt;= 0">Empty</DisplayString>

<DisplayString Condition="Data.ArrayNum &lt;= Data.ArrayMax">Num={Data.ArrayNum - NumFreeIndices}</DisplayString>

<Expand>

<IndexListItems Condition="Data.ArrayNum &gt; 0 &amp;&amp; Data.ArrayNum &lt;= Data.ArrayMax">

<Size>Data.ArrayNum</Size>

<ValueNode Condition="(AllocationFlags.AllocatorInstance.SecondaryData.Data != 0) &amp;&amp; (reinterpret\_cast&lt;uint32\*&gt;(AllocationFlags.AllocatorInstance.SecondaryData.Data)[$i/32]&gt;&gt;$i &amp; 1) != 0">\*(TSparseArray&lt;$T1,$T2&gt;::ElementType\*)((TSparseArray&lt;$T1,$T2&gt;::FElementOrFreeListLink\*)Data.AllocatorInstance.Data + $i)</ValueNode>

<ValueNode Condition="(AllocationFlags.AllocatorInstance.SecondaryData.Data == 0) &amp;&amp; (reinterpret\_cast&lt;uint32\*&gt;(AllocationFlags.AllocatorInstance.InlineData )[$i/32]&gt;&gt;$i &amp; 1) != 0">\*(TSparseArray&lt;$T1,$T2&gt;::ElementType\*)((TSparseArray&lt;$T1,$T2&gt;::FElementOrFreeListLink\*)Data.AllocatorInstance.Data + $i)</ValueNode>

<ValueNode Condition="(AllocationFlags.AllocatorInstance.SecondaryData.Data != 0) &amp;&amp; (reinterpret\_cast&lt;uint32\*&gt;(AllocationFlags.AllocatorInstance.SecondaryData.Data)[$i/32]&gt;&gt;$i &amp; 1) == 0">"Invalid",sb</ValueNode>

<ValueNode Condition="(AllocationFlags.AllocatorInstance.SecondaryData.Data == 0) &amp;&amp; (reinterpret\_cast&lt;uint32\*&gt;(AllocationFlags.AllocatorInstance.InlineData )[$i/32]&gt;&gt;$i &amp; 1) == 0">"Invalid",sb</ValueNode>

</IndexListItems>

</Expand>

</Type>

<!-- TBitArray visualizer -->

<Type Name="TBitArray&lt;\*&gt;">

<DisplayString Condition="NumBits == 0">Empty</DisplayString>

<DisplayString Condition="NumBits &lt; 0">Invalid</DisplayString>

<DisplayString Condition="NumBits &gt; 0">NumBits={NumBits}, MaxBits={MaxBits}</DisplayString>

<Expand>

<IndexListItems Condition="NumBits &gt; 0">

<Size>NumBits</Size>

<ValueNode Condition="(AllocatorInstance.SecondaryData.Data != 0) &amp;&amp; (reinterpret\_cast&lt;uint32\*&gt;(AllocatorInstance.SecondaryData.Data )[$i/32]&gt;&gt;$i &amp; 1) != 0">1</ValueNode>

<ValueNode Condition="(AllocatorInstance.SecondaryData.Data == 0) &amp;&amp; (reinterpret\_cast&lt;uint32\*&gt;(AllocatorInstance.InlineData )[$i/32]&gt;&gt;$i &amp; 1) != 0">1</ValueNode>

<ValueNode Condition="(AllocatorInstance.SecondaryData.Data != 0) &amp;&amp; (reinterpret\_cast&lt;uint32\*&gt;(AllocatorInstance.SecondaryData.Data )[$i/32]&gt;&gt;$i &amp; 1) == 0">0</ValueNode>

<ValueNode Condition="(AllocatorInstance.SecondaryData.Data == 0) &amp;&amp; (reinterpret\_cast&lt;uint32\*&gt;(AllocatorInstance.InlineData )[$i/32]&gt;&gt;$i &amp; 1) == 0">0</ValueNode>

</IndexListItems>

</Expand>

</Type>

<!-- TSharedPtr visualizer -->

<Type Name="TSharedPtr&lt;\*,\*&gt;">

<DisplayString Condition="Object == 0">Null</DisplayString>

<DisplayString Condition="Object != 0">Ptr={(void\*)Object}, SharedRefs={SharedReferenceCount.ReferenceController->SharedReferenceCount}, WeakRefs={SharedReferenceCount.ReferenceController->WeakReferenceCount}, Object={\*Object}</DisplayString>

<Expand>

<Item Condition="Object != 0" Name="[SharedReferenceCount]">SharedReferenceCount.ReferenceController->SharedReferenceCount</Item>

<Item Condition="Object != 0" Name="[WeakReferenceCount]">SharedReferenceCount.ReferenceController->WeakReferenceCount</Item>

<Item Condition="Object != 0" Name="[Ptr]">(void\*)Object</Item>

<ExpandedItem Condition="Object != 0">\*Object</ExpandedItem>

</Expand>

</Type>

<!-- TSharedRef visualizer -->

<Type Name="TSharedRef&lt;\*,\*&gt;">

<DisplayString Condition="Object != 0">Ptr={(void\*)Object}, SharedRefs={SharedReferenceCount.ReferenceController->SharedReferenceCount}, WeakRefs={SharedReferenceCount.ReferenceController->WeakReferenceCount}, Object={\*Object}</DisplayString>

<Expand>

<Item Condition="Object != 0" Name="[SharedReferenceCount]">SharedReferenceCount.ReferenceController->SharedReferenceCount</Item>

<Item Condition="Object != 0" Name="[WeakReferenceCount]">SharedReferenceCount.ReferenceController->WeakReferenceCount</Item>

<Item Condition="Object != 0" Name="[Ptr]">(void\*)Object</Item>

<ExpandedItem Condition="Object != 0">\*Object</ExpandedItem>

</Expand>

</Type>

<!-- TWeakPtr visualizer -->

<Type Name="TWeakPtr&lt;\*,\*&gt;">

<DisplayString Condition="Object == 0">Null</DisplayString>

<DisplayString Condition="WeakReferenceCount.ReferenceController->SharedReferenceCount == 0">Object has been destroyed</DisplayString>

<DisplayString Condition="Object != 0">Ptr={(void\*)Object}, SharedRefs={WeakReferenceCount.ReferenceController->SharedReferenceCount}, WeakRefs={WeakReferenceCount.ReferenceController->WeakReferenceCount}, Object={\*Object}</DisplayString>

<Expand>

<Item Condition="Object != 0" Name="[SharedReferenceCount]">WeakReferenceCount.ReferenceController->SharedReferenceCount</Item>

<Item Condition="Object != 0" Name="[WeakReferenceCount]">WeakReferenceCount.ReferenceController->WeakReferenceCount</Item>

<Item Condition="Object != 0 &amp;&amp; WeakReferenceCount.ReferenceController->SharedReferenceCount > 0" Name="[Ptr]">(void\*)Object</Item>

<ExpandedItem Condition="Object != 0 &amp;&amp; WeakReferenceCount.ReferenceController->SharedReferenceCount > 0">\*Object</ExpandedItem>

</Expand>

</Type>

<!-- TMapBase visualizer -->

<Type Name="TMapBase&lt;\*,\*,\*,\*&gt;">

<DisplayString Condition="Pairs.Elements.Data.ArrayNum - Pairs.Elements.NumFreeIndices &lt;= 0">Empty</DisplayString>

<DisplayString Condition="Pairs.Elements.Data.ArrayNum &lt;= Pairs.Elements.Data.ArrayMax" >Num={Pairs.Elements.Data.ArrayNum - Pairs.Elements.NumFreeIndices}</DisplayString>

<Expand>

<IndexListItems Condition="Pairs.Elements.Data.ArrayNum - Pairs.Elements.NumFreeIndices &gt; 0 &amp;&amp; Pairs.Elements.Data.ArrayNum &lt;= Pairs.Elements.Data.ArrayMax">

<Size>Pairs.Elements.Data.ArrayNum</Size>

<ValueNode Condition="(Pairs.Elements.AllocationFlags.AllocatorInstance.SecondaryData.Data != 0) &amp;&amp; (reinterpret\_cast&lt;uint32\*&gt;(Pairs.Elements.AllocationFlags.AllocatorInstance.SecondaryData.Data)[$i/32]&gt;&gt;$i &amp; 1) != 0">((TSetElement&lt;TTuple&lt;$T1,$T2&gt; &gt; \*)Pairs.Elements.Data.AllocatorInstance.Data)[$i].Value</ValueNode>

<ValueNode Condition="(Pairs.Elements.AllocationFlags.AllocatorInstance.SecondaryData.Data == 0) &amp;&amp; (reinterpret\_cast&lt;uint32\*&gt;(Pairs.Elements.AllocationFlags.AllocatorInstance.InlineData )[$i/32]&gt;&gt;$i &amp; 1) != 0">((TSetElement&lt;TTuple&lt;$T1,$T2&gt; &gt; \*)Pairs.Elements.Data.AllocatorInstance.Data)[$i].Value</ValueNode>

<ValueNode Condition="(Pairs.Elements.AllocationFlags.AllocatorInstance.SecondaryData.Data != 0) &amp;&amp; (reinterpret\_cast&lt;uint32\*&gt;(Pairs.Elements.AllocationFlags.AllocatorInstance.SecondaryData.Data)[$i/32]&gt;&gt;$i &amp; 1) == 0">"Invalid",sb</ValueNode>

<ValueNode Condition="(Pairs.Elements.AllocationFlags.AllocatorInstance.SecondaryData.Data == 0) &amp;&amp; (reinterpret\_cast&lt;uint32\*&gt;(Pairs.Elements.AllocationFlags.AllocatorInstance.InlineData )[$i/32]&gt;&gt;$i &amp; 1) == 0">"Invalid",sb</ValueNode>

</IndexListItems>

</Expand>

</Type>

<!-- TSet visualizer -->

<Type Name="TSet&lt;\*,\*,\*&gt;">

<DisplayString Condition="Elements.Data.ArrayNum - Elements.NumFreeIndices &lt;= 0">Empty</DisplayString>

<DisplayString Condition="Elements.Data.ArrayNum &lt;= Elements.Data.ArrayMax">Num={Elements.Data.ArrayNum - Elements.NumFreeIndices}</DisplayString>

<Expand>

<IndexListItems Condition="Elements.Data.ArrayNum - Elements.NumFreeIndices &gt; 0 &amp;&amp; Elements.Data.ArrayNum &lt;= Elements.Data.ArrayMax">

<Size>Elements.Data.ArrayNum</Size>

<ValueNode Condition="(Elements.AllocationFlags.AllocatorInstance.SecondaryData.Data != 0) &amp;&amp; (reinterpret\_cast&lt;uint32\*&gt;(Elements.AllocationFlags.AllocatorInstance.SecondaryData.Data)[$i/32]&gt;&gt;$i &amp; 1) != 0">((TSetElement &lt;$T1&gt; \*)Elements.Data.AllocatorInstance.Data)[$i].Value</ValueNode>

<ValueNode Condition="(Elements.AllocationFlags.AllocatorInstance.SecondaryData.Data == 0) &amp;&amp; (reinterpret\_cast&lt;uint32\*&gt;(Elements.AllocationFlags.AllocatorInstance.InlineData )[$i/32]&gt;&gt;$i &amp; 1) != 0">((TSetElement &lt;$T1&gt; \*)Elements.Data.AllocatorInstance.Data)[$i].Value</ValueNode>

<ValueNode Condition="(Elements.AllocationFlags.AllocatorInstance.SecondaryData.Data != 0) &amp;&amp; (reinterpret\_cast&lt;uint32\*&gt;(Elements.AllocationFlags.AllocatorInstance.SecondaryData.Data)[$i/32]&gt;&gt;$i &amp; 1) == 0">"Invalid",sb</ValueNode>

<ValueNode Condition="(Elements.AllocationFlags.AllocatorInstance.SecondaryData.Data == 0) &amp;&amp; (reinterpret\_cast&lt;uint32\*&gt;(Elements.AllocationFlags.AllocatorInstance.InlineData )[$i/32]&gt;&gt;$i &amp; 1) == 0">"Invalid",sb</ValueNode>

</IndexListItems>

</Expand>

</Type>

<!-- FWeakObjectPtr visualizer -->

<Type Name="FWeakObjectPtr">

<DisplayString Condition="ObjectSerialNumber &lt; 1">nullptr</DisplayString>

<DisplayString Condition="GObjectArrayForDebugVisualizers->Objects[ObjectIndex / 65536][ObjectIndex % 65536].SerialNumber != ObjectSerialNumber">STALE</DisplayString>

<DisplayString>{GObjectArrayForDebugVisualizers->Objects[ObjectIndex / 65536][ObjectIndex % 65536].Object}</DisplayString>

<Expand>

<ExpandedItem>GObjectArrayForDebugVisualizers->Objects[ObjectIndex / 65536][ObjectIndex % 65536].Object</ExpandedItem>

</Expand>

</Type>

<!-- TWeakObjectPtr<\*> visualizer -->

<Type Name="TWeakObjectPtr&lt;\*&gt;">

<DisplayString Condition="ObjectSerialNumber &lt; 1">nullptr</DisplayString>

<DisplayString Condition="GObjectArrayForDebugVisualizers->Objects[ObjectIndex / 65536][ObjectIndex % 65536].SerialNumber != ObjectSerialNumber">STALE</DisplayString>

<DisplayString>{($T1\*)GObjectArrayForDebugVisualizers->Objects[ObjectIndex / 65536][ObjectIndex % 65536].Object}</DisplayString>

<Expand>

<ExpandedItem>($T1\*)GObjectArrayForDebugVisualizers->Objects[ObjectIndex / 65536][ObjectIndex % 65536].Object</ExpandedItem>

</Expand>

</Type>

<!-- FSubobjectPtr visualizer -->

<Type Name="FSubobjectPtr">

<DisplayString>{Object}</DisplayString>

</Type>

<!-- FVertexID visualizer -->

<Type Name="FVertexID">

<DisplayString>{IDValue}</DisplayString>

</Type>

<!-- FVertexInstanceID visualizer -->

<Type Name="FVertexInstanceID">

<DisplayString>{IDValue}</DisplayString>

</Type>

<!-- FEdgeID visualizer -->

<Type Name="FEdgeID">

<DisplayString>{IDValue}</DisplayString>

</Type>

<!-- FPolygonID visualizer -->

<Type Name="FPolygonID">

<DisplayString>{IDValue}</DisplayString>

</Type>

<!-- FPolygonGroupID visualizer -->

<Type Name="FPolygonGroupID">

<DisplayString>{IDValue}</DisplayString>

</Type>

<!-- FTriangleID visualizer -->

<Type Name="FTriangleID">

<DisplayString>{IDValue}</DisplayString>

</Type>

<!-- TOptional visualizer -->

<Type Name="TOptional&lt;\*&gt;">

<DisplayString Condition="!bIsSet">Unset</DisplayString>

<DisplayString Condition="bIsSet">Set: {{{\*($T1\*)&amp;Value}}}</DisplayString>

<Expand>

<ExpandedItem Condition="bIsSet">\*($T1\*)&amp;Value</ExpandedItem>

</Expand>

</Type>

<!-- TInlineValue visualizer -->

<Type Name="TInlineValue&lt;\*&gt;">

<DisplayString Condition="!bIsValid">Null</DisplayString>

<DisplayString Condition="bIsValid &amp;&amp; bInline">{{{\*($T1\*)&amp;Data}}}</DisplayString>

<DisplayString Condition="bIsValid &amp;&amp; !bInline">{{{\*\*(($T1\*\*)&amp;Data)}}}</DisplayString>

<Expand>

<ExpandedItem Condition="bIsValid &amp;&amp; bInline">\*($T1\*)&amp;Data</ExpandedItem>

<ExpandedItem Condition="bIsValid &amp;&amp; !bInline">\*\*(($T1\*\*)&amp;Data)</ExpandedItem>

</Expand>

</Type>

<!-- TFunction visualizer -->

<Type Name="UE4Function\_Private::TDebugHelper&lt;\*&gt;">

<DisplayString>{\*Ptr}</DisplayString>

<Expand>

<Item Name="[Lambda]">\*Ptr</Item>

</Expand>

</Type>

<Type Name="TFunctionRef&lt;\*&gt;">

<DisplayString Condition="Callable">{DebugPtrStorage}</DisplayString>

<DisplayString Condition="!Callable">Unset</DisplayString>

<Expand>

<ExpandedItem Condition="Callable">DebugPtrStorage</ExpandedItem>

</Expand>

</Type>

<Type Name="TFunction&lt;\*&gt;">

<AlternativeType Name="TUniqueFunction&lt;\*&gt;"></AlternativeType>

<DisplayString Condition="Callable != 0">{DebugPtrStorage}</DisplayString>

<DisplayString Condition="Callable == 0">Unset</DisplayString>

<Expand>

<ExpandedItem Condition="Callable != 0">DebugPtrStorage</ExpandedItem>

</Expand>

</Type>

<!-- FGameplayTagContainer visualizer -->

<Type Name="FGameplayTagContainer">

<DisplayString Condition="GameplayTags.ArrayNum == 0">Empty</DisplayString>

<DisplayString Condition="GameplayTags.ArrayNum == 1">Tag={\*((FName\*)(GameplayTags.AllocatorInstance.Data))}</DisplayString>

<DisplayString Condition="GameplayTags.ArrayNum > 1">Num={GameplayTags.ArrayNum}</DisplayString>

<Expand>

<ArrayItems Condition="GameplayTags.ArrayNum &lt;= GameplayTags.ArrayMax">

<Size>GameplayTags.ArrayNum</Size>

<ValuePointer>((FName\*)(GameplayTags.AllocatorInstance.Data))</ValuePointer>

</ArrayItems>

</Expand>

</Type>

<!-- FActorRepList visualizer -->

<Type Name="FActorRepList">

<DisplayString >({Num}/{Max} {RefCount})</DisplayString>

<Expand>

<Item Name="[RefCount]">RefCount</Item>

<Item Name="[Num]">Num</Item>

<Item Name="[Max]">Max</Item>

<ArrayItems>

<Size>Num</Size>

<ValuePointer>((FActorRepListType\*)(Data))</ValuePointer>

</ArrayItems>

</Expand>

</Type>

<!-- TTuple visualizer -->

<Type Name="TTuple&lt;&gt;">

<DisplayString>{{}}</DisplayString>

<Expand>

</Expand>

</Type>

<Type Name="TTuple&lt;\*&gt;">

<DisplayString>{{{(\*(UE4Tuple\_Private::TTupleElement&lt;$T1,0&gt;\*)this).Value}}}</DisplayString>

<Expand>

<Item Name="[0]">((UE4Tuple\_Private::TTupleElement&lt;$T1,0&gt;\*)this)->Value</Item>

</Expand>

</Type>

<Type Name="TTuple&lt;\*,\*&gt;">

<DisplayString>{{{Key},{Value}}}</DisplayString>

<Expand>

<Item Name="[0:Key]">Key</Item>

<Item Name="[1:Value]">Value</Item>

</Expand>

</Type>

<Type Name="TTuple&lt;\*,\*,\*&gt;">

<DisplayString>{{{(\*(UE4Tuple\_Private::TTupleElement&lt;$T1,0&gt;\*)this).Value},{(\*(UE4Tuple\_Private::TTupleElement&lt;$T2,1&gt;\*)this).Value},{(\*(UE4Tuple\_Private::TTupleElement&lt;$T3,2&gt;\*)this).Value}}}</DisplayString>

<Expand>

<Item Name="[0]">((UE4Tuple\_Private::TTupleElement&lt;$T1,0&gt;\*)this)->Value</Item>

<Item Name="[1]">((UE4Tuple\_Private::TTupleElement&lt;$T2,1&gt;\*)this)->Value</Item>

<Item Name="[2]">((UE4Tuple\_Private::TTupleElement&lt;$T3,2&gt;\*)this)->Value</Item>

</Expand>

</Type>

<Type Name="TTuple&lt;\*,\*,\*,\*&gt;">

<DisplayString>{{{(\*(UE4Tuple\_Private::TTupleElement&lt;$T1,0&gt;\*)this).Value},{(\*(UE4Tuple\_Private::TTupleElement&lt;$T2,1&gt;\*)this).Value},{(\*(UE4Tuple\_Private::TTupleElement&lt;$T3,2&gt;\*)this).Value},{(\*(UE4Tuple\_Private::TTupleElement&lt;$T4,3&gt;\*)this).Value}}}</DisplayString>

<Expand>

<Item Name="[0]">((UE4Tuple\_Private::TTupleElement&lt;$T1,0&gt;\*)this)->Value</Item>

<Item Name="[1]">((UE4Tuple\_Private::TTupleElement&lt;$T2,1&gt;\*)this)->Value</Item>

<Item Name="[2]">((UE4Tuple\_Private::TTupleElement&lt;$T3,2&gt;\*)this)->Value</Item>

<Item Name="[3]">((UE4Tuple\_Private::TTupleElement&lt;$T4,3&gt;\*)this)->Value</Item>

</Expand>

</Type>

<Type Name="TTuple&lt;\*,\*,\*,\*,\*&gt;">

<DisplayString>{{{(\*(UE4Tuple\_Private::TTupleElement&lt;$T1,0&gt;\*)this).Value},{(\*(UE4Tuple\_Private::TTupleElement&lt;$T2,1&gt;\*)this).Value},{(\*(UE4Tuple\_Private::TTupleElement&lt;$T3,2&gt;\*)this).Value},{(\*(UE4Tuple\_Private::TTupleElement&lt;$T4,3&gt;\*)this).Value},{(\*(UE4Tuple\_Private::TTupleElement&lt;$T5,4&gt;\*)this).Value}}}</DisplayString>

<Expand>

<Item Name="[0]">((UE4Tuple\_Private::TTupleElement&lt;$T1,0&gt;\*)this)->Value</Item>

<Item Name="[1]">((UE4Tuple\_Private::TTupleElement&lt;$T2,1&gt;\*)this)->Value</Item>

<Item Name="[2]">((UE4Tuple\_Private::TTupleElement&lt;$T3,2&gt;\*)this)->Value</Item>

<Item Name="[3]">((UE4Tuple\_Private::TTupleElement&lt;$T4,3&gt;\*)this)->Value</Item>

<Item Name="[4]">((UE4Tuple\_Private::TTupleElement&lt;$T5,4&gt;\*)this)->Value</Item>

</Expand>

</Type>

<Type Name="TTuple&lt;\*,\*,\*,\*,\*,\*&gt;">

<DisplayString>{{{(\*(UE4Tuple\_Private::TTupleElement&lt;$T1,0&gt;\*)this).Value},{(\*(UE4Tuple\_Private::TTupleElement&lt;$T2,1&gt;\*)this).Value},{(\*(UE4Tuple\_Private::TTupleElement&lt;$T3,2&gt;\*)this).Value},{(\*(UE4Tuple\_Private::TTupleElement&lt;$T4,3&gt;\*)this).Value},{(\*(UE4Tuple\_Private::TTupleElement&lt;$T5,4&gt;\*)this).Value},{(\*(UE4Tuple\_Private::TTupleElement&lt;$T6,5&gt;\*)this).Value}}}</DisplayString>

<Expand>

<Item Name="[0]">((UE4Tuple\_Private::TTupleElement&lt;$T1,0&gt;\*)this)->Value</Item>

<Item Name="[1]">((UE4Tuple\_Private::TTupleElement&lt;$T2,1&gt;\*)this)->Value</Item>

<Item Name="[2]">((UE4Tuple\_Private::TTupleElement&lt;$T3,2&gt;\*)this)->Value</Item>

<Item Name="[3]">((UE4Tuple\_Private::TTupleElement&lt;$T4,3&gt;\*)this)->Value</Item>

<Item Name="[4]">((UE4Tuple\_Private::TTupleElement&lt;$T5,4&gt;\*)this)->Value</Item>

<Item Name="[5]">((UE4Tuple\_Private::TTupleElement&lt;$T6,5&gt;\*)this)->Value</Item>

</Expand>

</Type>

<Type Name="TTuple&lt;\*,\*,\*,\*,\*,\*,\*&gt;">

<DisplayString>{{{(\*(UE4Tuple\_Private::TTupleElement&lt;$T1,0&gt;\*)this).Value},{(\*(UE4Tuple\_Private::TTupleElement&lt;$T2,1&gt;\*)this).Value},{(\*(UE4Tuple\_Private::TTupleElement&lt;$T3,2&gt;\*)this).Value},{(\*(UE4Tuple\_Private::TTupleElement&lt;$T4,3&gt;\*)this).Value},{(\*(UE4Tuple\_Private::TTupleElement&lt;$T5,4&gt;\*)this).Value},{(\*(UE4Tuple\_Private::TTupleElement&lt;$T6,5&gt;\*)this).Value},{(\*(UE4Tuple\_Private::TTupleElement&lt;$T7,6&gt;\*)this).Value}}}</DisplayString>

<Expand>

<Item Name="[0]">((UE4Tuple\_Private::TTupleElement&lt;$T1,0&gt;\*)this)->Value</Item>

<Item Name="[1]">((UE4Tuple\_Private::TTupleElement&lt;$T2,1&gt;\*)this)->Value</Item>

<Item Name="[2]">((UE4Tuple\_Private::TTupleElement&lt;$T3,2&gt;\*)this)->Value</Item>

<Item Name="[3]">((UE4Tuple\_Private::TTupleElement&lt;$T4,3&gt;\*)this)->Value</Item>

<Item Name="[4]">((UE4Tuple\_Private::TTupleElement&lt;$T5,4&gt;\*)this)->Value</Item>

<Item Name="[5]">((UE4Tuple\_Private::TTupleElement&lt;$T6,5&gt;\*)this)->Value</Item>

<Item Name="[6]">((UE4Tuple\_Private::TTupleElement&lt;$T7,6&gt;\*)this)->Value</Item>

</Expand>

</Type>

<!-- FDelegateBase visualizer -->

<Type Name="TBaseStaticDelegateInstance&lt;\*&gt;">

<DisplayString>{StaticFuncPtr}</DisplayString>

<Expand>

<Item Name="[StaticFuncPtr]">StaticFuncPtr</Item>

<Item Name="[Payload]">Payload</Item>

<Item Name="[Handle]">Handle</Item>

</Expand>

</Type>

<Type Name="TBaseFunctorDelegateInstance&lt;\*&gt;">

<DisplayString>{Functor}</DisplayString>

<Expand>

<Item Name="[Functor]">Functor</Item>

<Item Name="[Payload]">Payload</Item>

<Item Name="[Handle]">Handle</Item>

</Expand>

</Type>

<Type Name="TBaseRawMethodDelegateInstance&lt;\*,\*,\*&gt;">

<DisplayString>{UserObject}</DisplayString>

<Expand>

<Item Name="[UserObject]">UserObject</Item>

<Item Name="[MethodPtr]">MethodPtr</Item>

<Item Name="[Payload]">Payload</Item>

<Item Name="[Handle]">Handle</Item>

</Expand>

</Type>

<Type Name="TBaseSPMethodDelegateInstance&lt;\*,\*,\*&gt;">

<DisplayString>{UserObject}</DisplayString>

<Expand>

<Item Name="[UserObject]">UserObject</Item>

<Item Name="[MethodPtr]">MethodPtr</Item>

<Item Name="[Payload]">Payload</Item>

<Item Name="[Handle]">Handle</Item>

</Expand>

</Type>

<Type Name="TBaseUObjectMethodDelegateInstance&lt;\*,\*,\*&gt;">

<DisplayString>{UserObject}</DisplayString>

<Expand>

<Item Name="[UserObject]">UserObject</Item>

<Item Name="[MethodPtr]">MethodPtr</Item>

<Item Name="[Payload]">Payload</Item>

<Item Name="[Handle]">Handle</Item>

</Expand>

</Type>

<Type Name="TBaseUFunctionDelegateInstance&lt;\*,\*&gt;">

<DisplayString>{UserObjectPtr}</DisplayString>

<Expand>

<Item Name="[UserObject]">UserObjectPtr</Item>

<Item Name="[FunctionName]">FunctionName</Item>

<Item Name="[Payload]">Payload</Item>

<Item Name="[Handle]">Handle</Item>

</Expand>

</Type>

<Type Name="FDelegateBase">

<DisplayString Condition="DelegateSize == 0">Unbound</DisplayString>

<DisplayString Condition="DelegateSize &lt;= 2">{\*((IDelegateInstance\*)(DelegateAllocator.InlineData))}</DisplayString>

<DisplayString Condition="DelegateSize &gt; 2">{\*((IDelegateInstance\*)(DelegateAllocator.SecondaryData.Data))}</DisplayString>

<Expand>

<ExpandedItem Condition="DelegateSize == 0">DelegateAllocator</ExpandedItem>

<ExpandedItem Condition="DelegateSize &lt;= 2">\*((IDelegateInstance\*)(DelegateAllocator.InlineData))</ExpandedItem>

<ExpandedItem Condition="DelegateSize &gt; 2">\*((IDelegateInstance\*)(DelegateAllocator.SecondaryData.Data))</ExpandedItem>

</Expand>

</Type>

<Type Name="TAtomic&lt;\*&gt;">

<DisplayString>{Element}</DisplayString>

<Expand>

<ExpandedItem>Element</ExpandedItem>

</Expand>

</Type>

<!-- TRange visualizer -->

<Type Name="TRange&lt;\*&gt;">

<DisplayString Condition="LowerBound.Type.Value == ERangeBoundTypes::Open &amp;&amp; UpperBound.Type.Value == ERangeBoundTypes::Open">[-&#8734;, +&#8734;]</DisplayString>

<DisplayString Condition="LowerBound.Type.Value == ERangeBoundTypes::Open &amp;&amp; UpperBound.Type.Value == ERangeBoundTypes::Inclusive">[-&#8734;, {UpperBound.Value}]</DisplayString>

<DisplayString Condition="LowerBound.Type.Value == ERangeBoundTypes::Open &amp;&amp; UpperBound.Type.Value == ERangeBoundTypes::Exclusive">[-&#8734;, {UpperBound.Value})</DisplayString>

<DisplayString Condition="LowerBound.Type.Value == ERangeBoundTypes::Inclusive &amp;&amp; UpperBound.Type.Value == ERangeBoundTypes::Open">[{LowerBound.Value}, +&#8734;)</DisplayString>

<DisplayString Condition="LowerBound.Type.Value == ERangeBoundTypes::Inclusive &amp;&amp; UpperBound.Type.Value == ERangeBoundTypes::Inclusive">[{LowerBound.Value}, {UpperBound.Value}]</DisplayString>

<DisplayString Condition="LowerBound.Type.Value == ERangeBoundTypes::Inclusive &amp;&amp; UpperBound.Type.Value == ERangeBoundTypes::Exclusive">[{LowerBound.Value}, {UpperBound.Value})</DisplayString>

<DisplayString Condition="LowerBound.Type.Value == ERangeBoundTypes::Exclusive &amp;&amp; UpperBound.Type.Value == ERangeBoundTypes::Open">({LowerBound.Value}, +&#8734;]</DisplayString>

<DisplayString Condition="LowerBound.Type.Value == ERangeBoundTypes::Exclusive &amp;&amp; UpperBound.Type.Value == ERangeBoundTypes::Inclusive">({LowerBound.Value}, {UpperBound.Value}]</DisplayString>

<DisplayString Condition="LowerBound.Type.Value == ERangeBoundTypes::Exclusive &amp;&amp; UpperBound.Type.Value == ERangeBoundTypes::Exclusive">({LowerBound.Value}, {UpperBound.Value})</DisplayString>

</Type>

<!-- FFrameNumber visualizer -->

<Type Name="FFrameNumber">

<DisplayString>{Value}</DisplayString>

</Type>

<!-- FFrameTime visualizer -->

<Type Name="FFrameTime">

<DisplayString Condition="SubFrame == 0.0">{FrameNumber}</DisplayString>

<DisplayString Condition="SubFrame != 0.0">{(double)(FrameNumber.Value)+SubFrame}</DisplayString>

</Type>

<!-- FRHICommandList visualizer -->

<Type Name="FRHICommandBase">

<DisplayString>{{ RHI Command -> { this->\_\_vfptr[0] } }}</DisplayString>

</Type>

<Type Name="FRHICommandList">

<Expand>

<LinkedListItems>

<HeadPointer>Root</HeadPointer>

<NextPointer>Next</NextPointer>

<ValueNode>this</ValueNode>

</LinkedListItems>

</Expand>

</Type>

</AutoVisualizer>

[/Script/EditorStyle.EditorStyleSettings]

ColorVisionDeficiencyPreviewType=NormalVision

ColorVisionDeficiencySeverity=3

bColorVisionDeficiencyCorrection=False

bColorVisionDeficiencyCorrectionPreviewWithDeficiency=False

SelectionColor=(R=0.728000,G=0.364000,B=0.003000,A=1.000000)

PressedSelectionColor=(R=0.701000,G=0.225000,B=0.003000,A=1.000000)

InactiveSelectionColor=(R=0.250000,G=0.250000,B=0.250000,A=1.000000)

KeyboardFocusColor=(R=0.000000,G=0.000000,B=0.000000,A=0.000000)

EditorWindowBackgroundColor=(R=1.000000,G=1.000000,B=1.000000,A=1.000000)

EditorMainWindowBackgroundOverride=(ImageSize=(X=32.000000,Y=32.000000),Margin=(Left=0.000000,Top=0.000000,Right=0.000000,Bottom=0.000000),Tint=(R=1.000000,G=1.000000,B=1.000000,A=1.000000),TintColor=(SpecifiedColor=(R=1.000000,G=1.000000,B=1.000000,A=1.000000),ColorUseRule=UseColor\_Specified),ResourceObject=None,ResourceName="",UVRegion=(Min=(X=0.000000,Y=0.000000),Max=(X=0.000000,Y=0.000000),bIsValid=0),DrawAs=Image,Tiling=NoTile,Mirroring=NoMirror,ImageType=NoImage,bIsDynamicallyLoaded=False,bHasUObject=False)

EditorChildWindowBackgroundOverride=(ImageSize=(X=32.000000,Y=32.000000),Margin=(Left=0.000000,Top=0.000000,Right=0.000000,Bottom=0.000000),Tint=(R=1.000000,G=1.000000,B=1.000000,A=1.000000),TintColor=(SpecifiedColor=(R=1.000000,G=1.000000,B=1.000000,A=1.000000),ColorUseRule=UseColor\_Specified),ResourceObject=None,ResourceName="",UVRegion=(Min=(X=0.000000,Y=0.000000),Max=(X=0.000000,Y=0.000000),bIsValid=0),DrawAs=Image,Tiling=NoTile,Mirroring=NoMirror,ImageType=NoImage,bIsDynamicallyLoaded=False,bHasUObject=False)

bResetEditorWindowBackgroundSettings=False

bUseSmallToolBarIcons=False

bUseGrid=True

RegularColor=(R=0.035000,G=0.035000,B=0.035000,A=1.000000)

RuleColor=(R=0.008000,G=0.008000,B=0.008000,A=1.000000)

CenterColor=(R=0.000000,G=0.000000,B=0.000000,A=1.000000)

GridSnapSize=16

bEnableWindowAnimations=False

bShowFriendlyNames=True

bExpandConfigurationMenus=False

bShowProjectMenus=True

bShowLaunchMenus=True

LogBackgroundColor=(R=0.015996,G=0.015996,B=0.015996,A=1.000000)

LogSelectionBackgroundColor=(R=0.008023,G=0.008023,B=0.008023,A=1.000000)

LogNormalColor=(R=0.720000,G=0.720000,B=0.720000,A=1.000000)

LogCommandColor=(R=0.033105,G=0.723055,B=0.033105,A=1.000000)

LogWarningColor=(R=0.921875,G=0.691406,B=0.000000,A=1.000000)

LogErrorColor=(R=1.000000,G=0.052083,B=0.060957,A=1.000000)

bShowAllAdvancedDetails=False

bShowHiddenPropertiesWhilePlaying=False

LogFontSize=9

LogTimestampMode=None

bPromoteOutputLogWarningsDuringPIE=False

AssetEditorOpenLocation=MainWindow

bEnableColorizedEditorTabs=True

[/Script/IOSRuntimeSettings.IOSRuntimeSettings]

bSupportsLandscapeLeftOrientation=True

PreferredLandscapeOrientation=LandscapeLeft

bSupportsPortraitOrientation=False

bSupportsUpsideDownOrientation=False

[/Script/EngineSettings.GameMapsSettings]

GlobalDefaultGameMode=/Game/Bluprint/BP\_GameMode.BP\_GameMode\_C

EditorStartupMap=/Game/Maps/Main.Main

GameDefaultMap=/Game/Maps/Main.Main

[/Script/Engine.RendererSettings]

r.DefaultFeature.AutoExposure=False

[/Script/Engine.PhysicsSettings]

DefaultGravityZ=-980.000000

DefaultTerminalVelocity=4000.000000

DefaultFluidFriction=0.300000

SimulateScratchMemorySize=262144

RagdollAggregateThreshold=4

TriangleMeshTriangleMinAreaThreshold=5.000000

bEnableShapeSharing=False

bEnablePCM=True

bEnableStabilization=False

bWarnMissingLocks=True

bEnable2DPhysics=False

PhysicErrorCorrection=(PingExtrapolation=0.100000,PingLimit=100.000000,ErrorPerLinearDifference=1.000000,ErrorPerAngularDifference=1.000000,MaxRestoredStateError=1.000000,MaxLinearHardSnapDistance=400.000000,PositionLerp=0.000000,AngleLerp=0.400000,LinearVelocityCoefficient=100.000000,AngularVelocityCoefficient=10.000000,ErrorAccumulationSeconds=0.500000,ErrorAccumulationDistanceSq=15.000000,ErrorAccumulationSimilarity=100.000000)

LockedAxis=Invalid

DefaultDegreesOfFreedom=Full3D

BounceThresholdVelocity=200.000000

FrictionCombineMode=Average

RestitutionCombineMode=Average

MaxAngularVelocity=3600.000000

MaxDepenetrationVelocity=0.000000

ContactOffsetMultiplier=0.020000

MinContactOffset=2.000000

MaxContactOffset=8.000000

bSimulateSkeletalMeshOnDedicatedServer=True

DefaultShapeComplexity=CTF\_UseSimpleAndComplex

bDefaultHasComplexCollision=True

bSuppressFaceRemapTable=False

bSupportUVFromHitResults=False

bDisableActiveActors=False

bDisableKinematicStaticPairs=False

bDisableKinematicKinematicPairs=False

bDisableCCD=False

bEnableEnhancedDeterminism=False

MaxPhysicsDeltaTime=0.033333

bSubstepping=False

bSubsteppingAsync=False

MaxSubstepDeltaTime=0.016667

MaxSubsteps=6

SyncSceneSmoothingFactor=0.000000

InitialAverageFrameRate=0.016667

PhysXTreeRebuildRate=10

DefaultBroadphaseSettings=(bUseMBPOnClient=False,bUseMBPOnServer=False,MBPBounds=(Min=(X=0.000000,Y=0.000000,Z=0.000000),Max=(X=0.000000,Y=0.000000,Z=0.000000),IsValid=0),MBPNumSubdivs=2)

[/Script/EngineSettings.GeneralProjectSettings]

ProjectID=F6969B424F3CE902F3EA319F61D6D498

[/Script/Engine.InputSettings]

-AxisConfig=(AxisKeyName="Gamepad\_LeftX",AxisProperties=(DeadZone=0.25,Exponent=1.f,Sensitivity=1.f))

-AxisConfig=(AxisKeyName="Gamepad\_LeftY",AxisProperties=(DeadZone=0.25,Exponent=1.f,Sensitivity=1.f))

-AxisConfig=(AxisKeyName="Gamepad\_RightX",AxisProperties=(DeadZone=0.25,Exponent=1.f,Sensitivity=1.f))

-AxisConfig=(AxisKeyName="Gamepad\_RightY",AxisProperties=(DeadZone=0.25,Exponent=1.f,Sensitivity=1.f))

-AxisConfig=(AxisKeyName="MouseX",AxisProperties=(DeadZone=0.f,Exponent=1.f,Sensitivity=0.07f))

-AxisConfig=(AxisKeyName="MouseY",AxisProperties=(DeadZone=0.f,Exponent=1.f,Sensitivity=0.07f))

+AxisConfig=(AxisKeyName="Gamepad\_LeftX",AxisProperties=(DeadZone=0.250000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="Gamepad\_LeftY",AxisProperties=(DeadZone=0.250000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="Gamepad\_RightX",AxisProperties=(DeadZone=0.250000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="Gamepad\_RightY",AxisProperties=(DeadZone=0.250000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="MouseX",AxisProperties=(DeadZone=0.000000,Sensitivity=0.070000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="MouseY",AxisProperties=(DeadZone=0.000000,Sensitivity=0.070000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="MotionController\_Left\_Thumbstick\_Z",AxisProperties=(DeadZone=0.000000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="MotionController\_Right\_Thumbstick\_Z",AxisProperties=(DeadZone=0.000000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="MouseWheelAxis",AxisProperties=(DeadZone=0.000000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="Gamepad\_LeftTriggerAxis",AxisProperties=(DeadZone=0.000000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="Gamepad\_RightTriggerAxis",AxisProperties=(DeadZone=0.000000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="MotionController\_Left\_Thumbstick\_X",AxisProperties=(DeadZone=0.000000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="MotionController\_Left\_Thumbstick\_Y",AxisProperties=(DeadZone=0.000000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="MotionController\_Left\_TriggerAxis",AxisProperties=(DeadZone=0.000000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="MotionController\_Left\_Grip1Axis",AxisProperties=(DeadZone=0.000000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="MotionController\_Left\_Grip2Axis",AxisProperties=(DeadZone=0.000000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="MotionController\_Right\_Thumbstick\_X",AxisProperties=(DeadZone=0.000000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="MotionController\_Right\_Thumbstick\_Y",AxisProperties=(DeadZone=0.000000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="MotionController\_Right\_TriggerAxis",AxisProperties=(DeadZone=0.000000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="MotionController\_Right\_Grip1Axis",AxisProperties=(DeadZone=0.000000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="MotionController\_Right\_Grip2Axis",AxisProperties=(DeadZone=0.000000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="Gamepad\_Special\_Left\_X",AxisProperties=(DeadZone=0.000000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

+AxisConfig=(AxisKeyName="Gamepad\_Special\_Left\_Y",AxisProperties=(DeadZone=0.000000,Sensitivity=1.000000,Exponent=1.000000,bInvert=False))

bAltEnterTogglesFullscreen=True

bF11TogglesFullscreen=True

bUseMouseForTouch=False

bEnableMouseSmoothing=True

bEnableFOVScaling=True

bCaptureMouseOnLaunch=True

bDefaultViewportMouseLock=False

bAlwaysShowTouchInterface=False

bShowConsoleOnFourFingerTap=True

bEnableGestureRecognizer=False

bUseAutocorrect=False

DefaultViewportMouseCaptureMode=CapturePermanently\_IncludingInitialMouseDown

DefaultViewportMouseLockMode=LockOnCapture

FOVScale=0.011110

DoubleClickTime=0.200000

+ActionMappings=(ActionName="PrimaryFire",bShift=False,bCtrl=False,bAlt=False,bCmd=False,Key=LeftMouseButton)

+ActionMappings=(ActionName="SecondaryFire",bShift=False,bCtrl=False,bAlt=False,bCmd=False,Key=RightMouseButton)

+AxisMappings=(AxisName="MoveUp",Scale=1.000000,Key=Up)

+AxisMappings=(AxisName="MoveUp",Scale=-1.000000,Key=Down)

+AxisMappings=(AxisName="MoveUp",Scale=1.000000,Key=W)

+AxisMappings=(AxisName="MoveUp",Scale=-1.000000,Key=S)

+AxisMappings=(AxisName="MoveRight",Scale=-1.000000,Key=Left)

+AxisMappings=(AxisName="MoveRight",Scale=1.000000,Key=Right)

+AxisMappings=(AxisName="MoveRight",Scale=-1.000000,Key=A)

+AxisMappings=(AxisName="MoveRight",Scale=1.000000,Key=D)

DefaultTouchInterface=/Engine/MobileResources/HUD/DefaultVirtualJoysticks.DefaultVirtualJoysticks

ConsoleKey=None

-ConsoleKeys=Tilde

+ConsoleKeys=Tilde