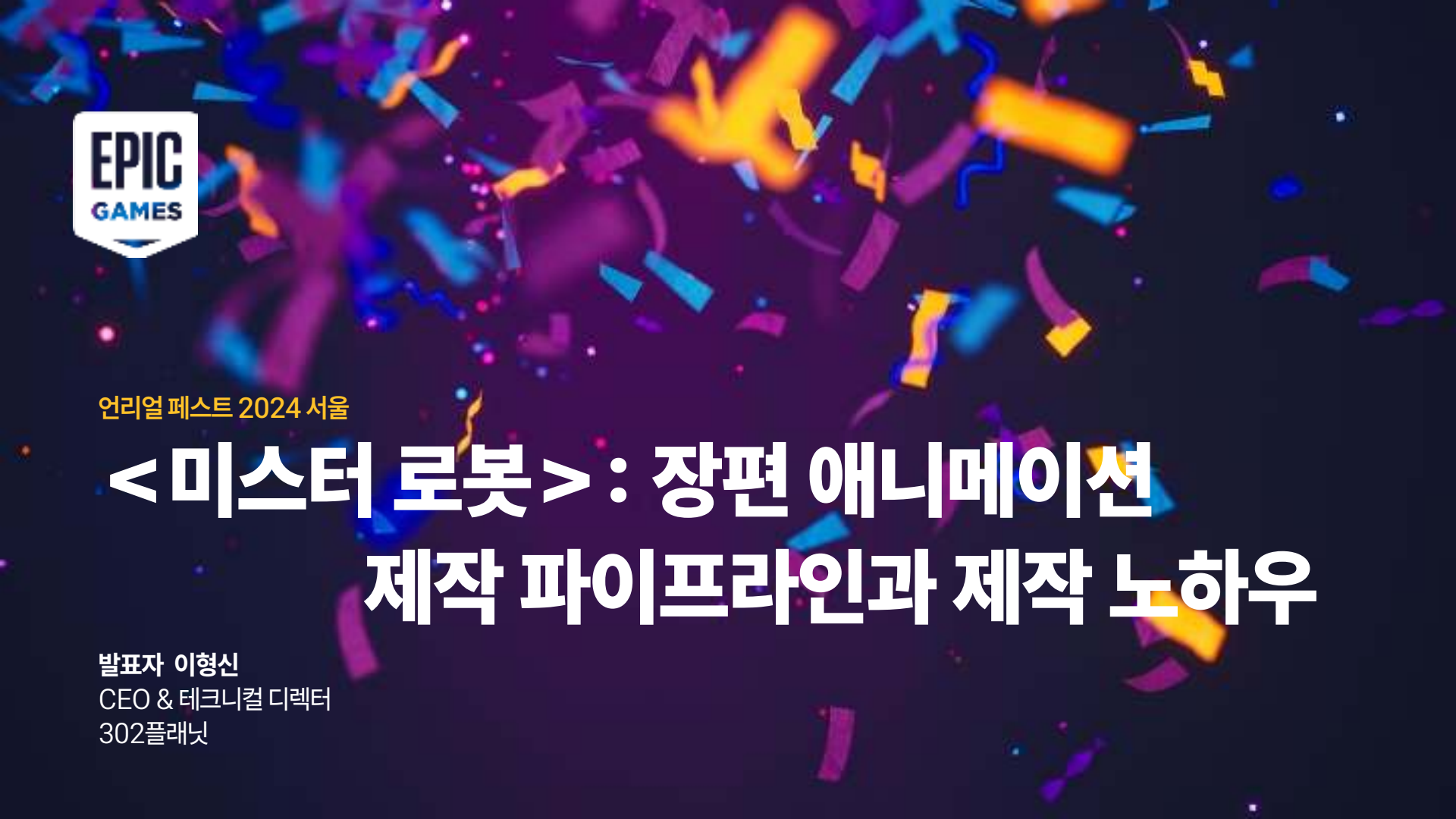


언리얼 페스트 2024 서울

<미스터 로봇> : 장편 애니메이션 제작 파이프라인과 제작 노하우

발표자 이형신
CEO & 테크니컬 디렉터
302플래닛

목차

Section 1

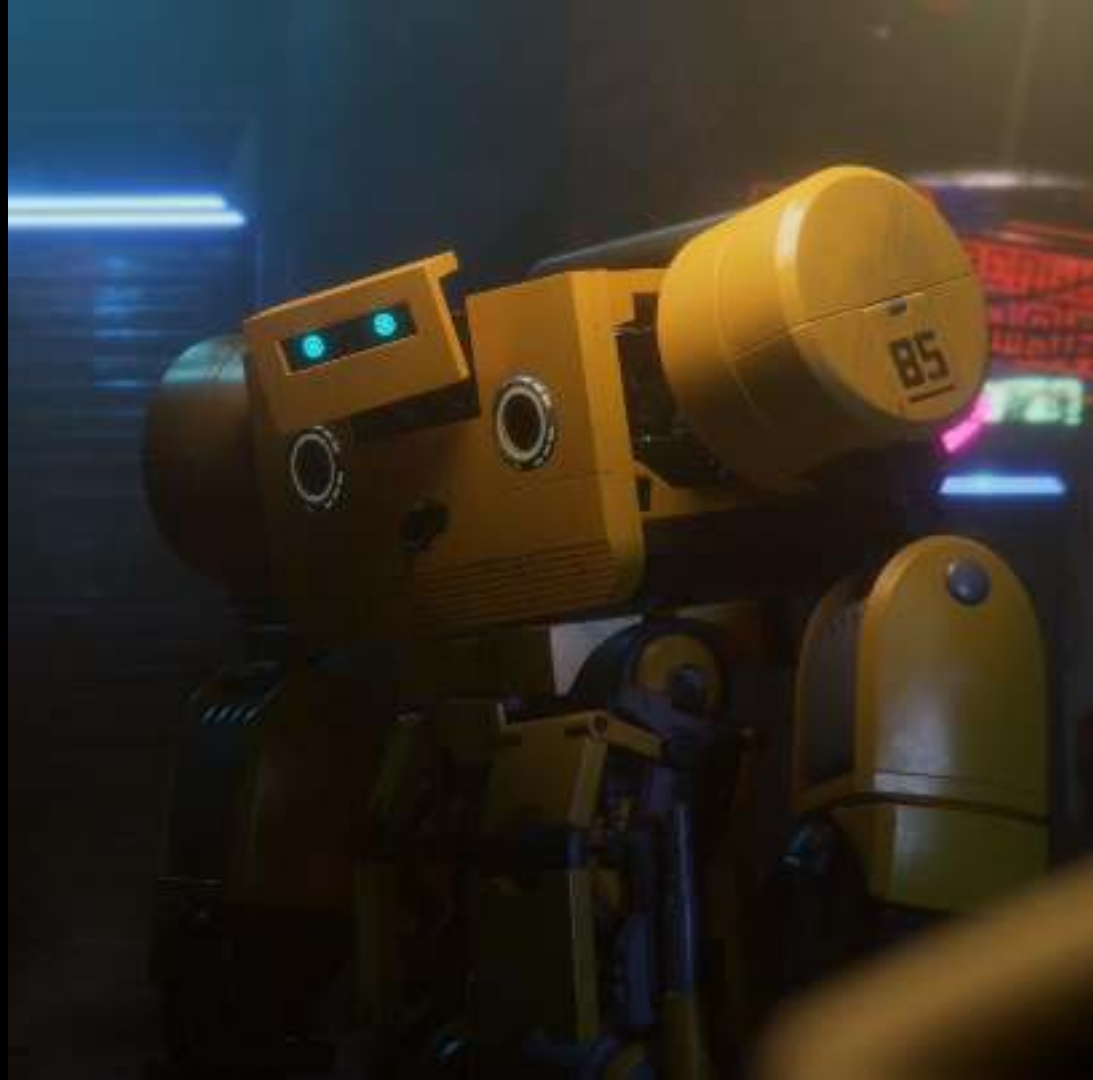
<미스터 로봇> 작품 소개 및 회사 소개

- <미스터 로봇> 작품 소개
- 302플래닛 회사 소개

미스터 로봇

- 장편 애니메이션
- SF 액션
- 12세 이상
- 2024년 11월 극장 개봉 예정
- 302플래닛, 이대애니메이션스튜디오 공동제작

EPIC MegaGrants Recipient



302planet

- 3D 애니메이션 전문 기업
- 3D 제작 공정관리 프로그램(Starfish) 자체 개발(특허 보유)
- <스트레스 제로> 장편 애니메이션 제작
  - 2021년 2월 극장 개봉
- 수십 편의 애니메이션 제작 참여

Section 2

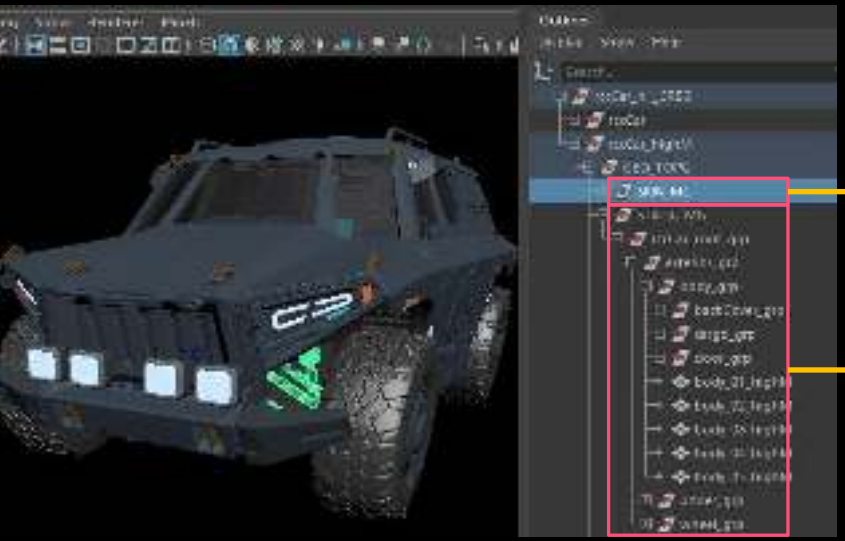
애니메이션 제작 파이프라인 살펴보기

- Maya to Unreal Engine
- 공정관리 툴과 연동하기
- Sequence 구성
- Asset Class 활용

Maya to Unreal Engine

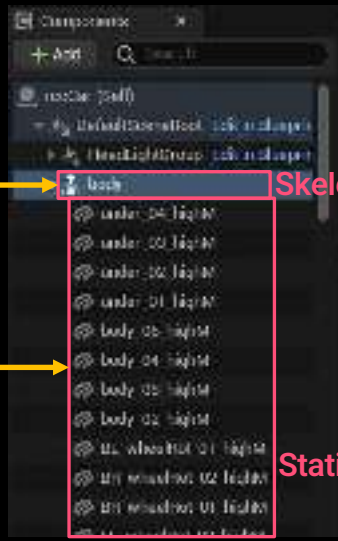
Skeletal Mesh + Static Mesh
Fbx 데이타 제어하기

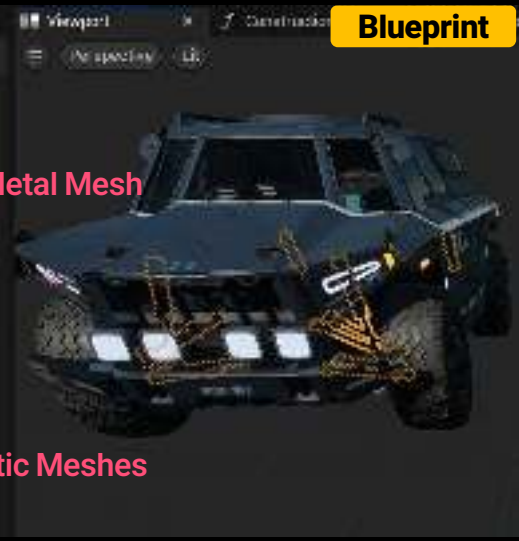
Maya to Unreal Engine(Asset)



Fbx

Modify Fbx





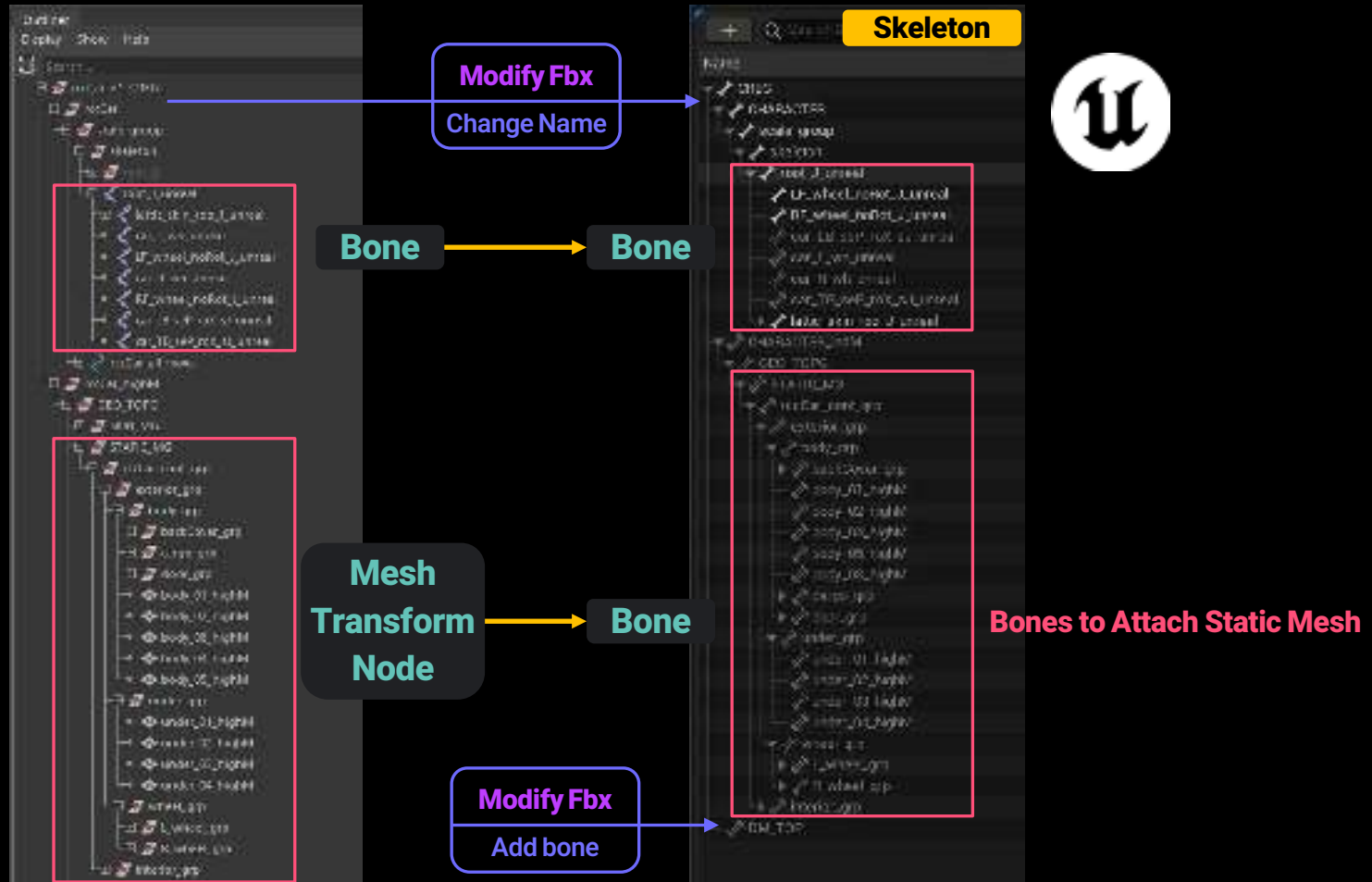
Skin된 Mesh들과 Skin되지 않은 Mesh들을 하나의 Fbx로 Export

Skin된 Mesh가 없을 경우 skin된 더미 Mesh를 만들어준다
(Modify Fbx)

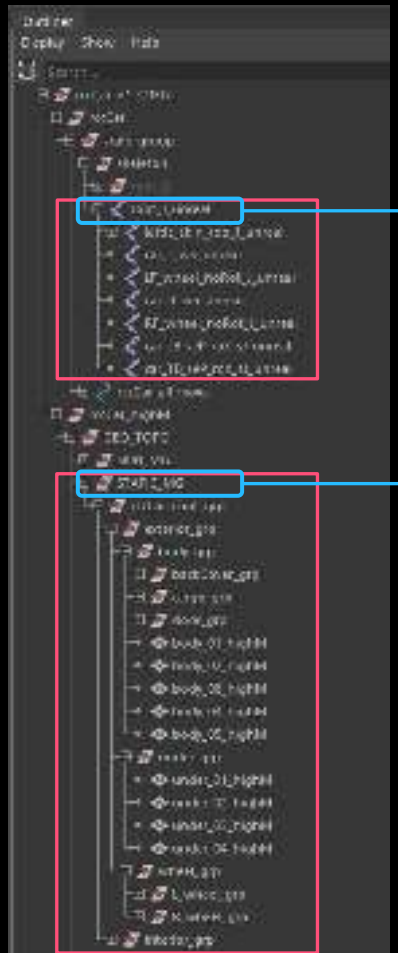
Fbx를 한 번은 Skeletal Mesh로, 또 한 번은 Static Mesh로 Import

Static Mesh로 Import할때 Fbx의 property 값을 이용해서 Skeletal Mesh뽑냄

Maya to Unreal Engine(Asset)



Maya to Unreal Engine(Animation)



Select

Fbx Export

Export Animation Only
(without Mesh)

Include Children

Modify Fbx

Change Name

Set Skeleton Attr
(Null ➡ Skeleton)

Animation Sequence

FBX Python SDK 활용

Export

Fbx

Modify Fbx

Import

Get Info From Fbx

Change Node Name
Maya에 reference로 들어와 있어
이름 바꾸기 힘든 Node 이름 변경

Add Node(Bone)
Blueprint 작업에 필요한 더미 bone 생성

Set Node Property(Metadata)
Static Mesh를 구분하기 위한 Property 추가

Set Skeleton Attribute
Null type의 node들을 skeleton type으로 변경

Add dummy Mesh and Skin to Root
Skin된 Mesh가 없을때 Skeletal Mesh를 만들기 위해
Dummy Mesh 만들어서 skin 추가

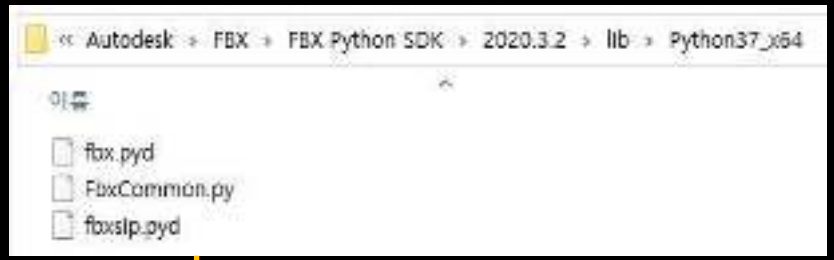
Get Creation Time
Unreal Engine 어셋에 metadata로 저장
어셋 업데이트가 필요한지 체크할 때 사용

Get Start, End Frame
Export 당시 Start, End Frame정보 확인

FBX Python SDK 설치

AUTODESK
Platform Services

FBX Python SDK



Import python module

GitHub

Meatplowz / FBX_Scene.py
Last active 4 months ago
This is a wrapper FBX class useful in accessing and modifying the FBX Scene

Fbx 수정에 필요한 기본 함수들 포함

필요 기능들 추가 및 수정해서 사용

공정관리 툴과 Unreal Engine 연동하기

Python Remote Execution

Remote Control Unreal Engine

공정 관리
프로그램

❶Send Python Command

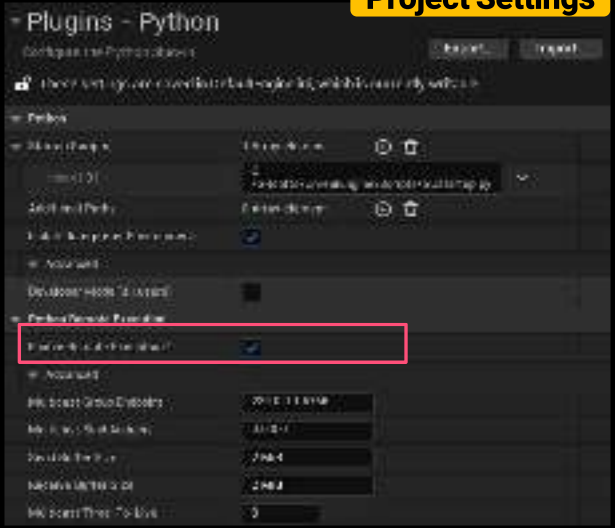
❸Get Python Print Result

❷Execute Python Command

remote_execution.py 사용

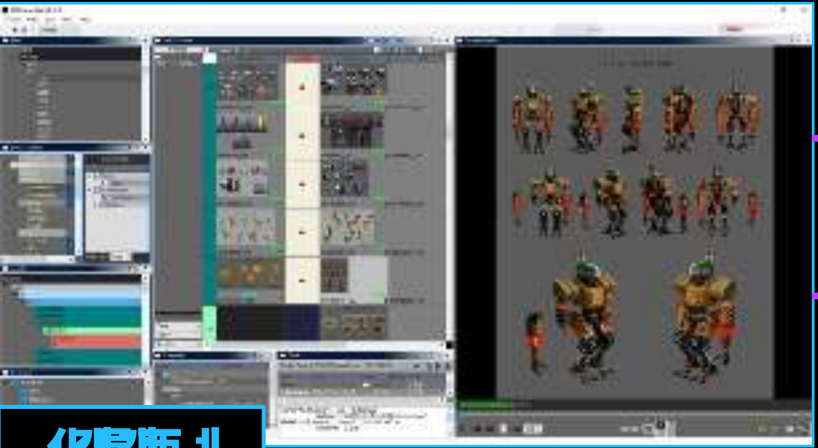
경로 : …\Epic Games\UE_5.3\Engine
\Plugins\Experimental\PythonScriptPlugin\Content\Python

Project Settings

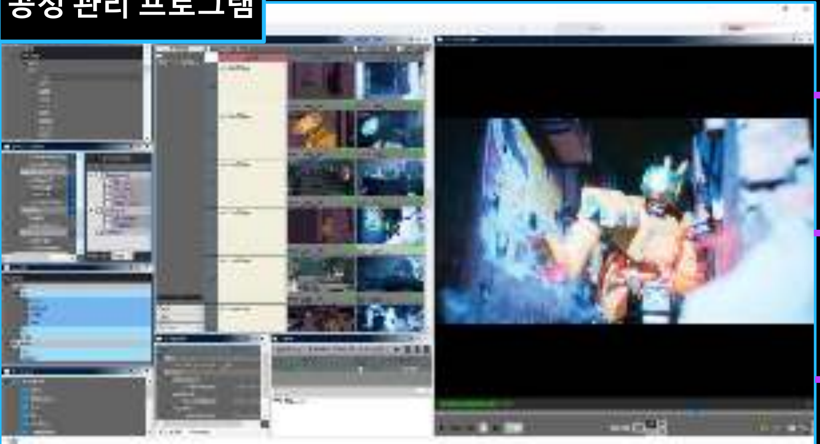


Python -> Enable Remote Execution

Starfish to Unreal Engine



공정 관리 프로그램



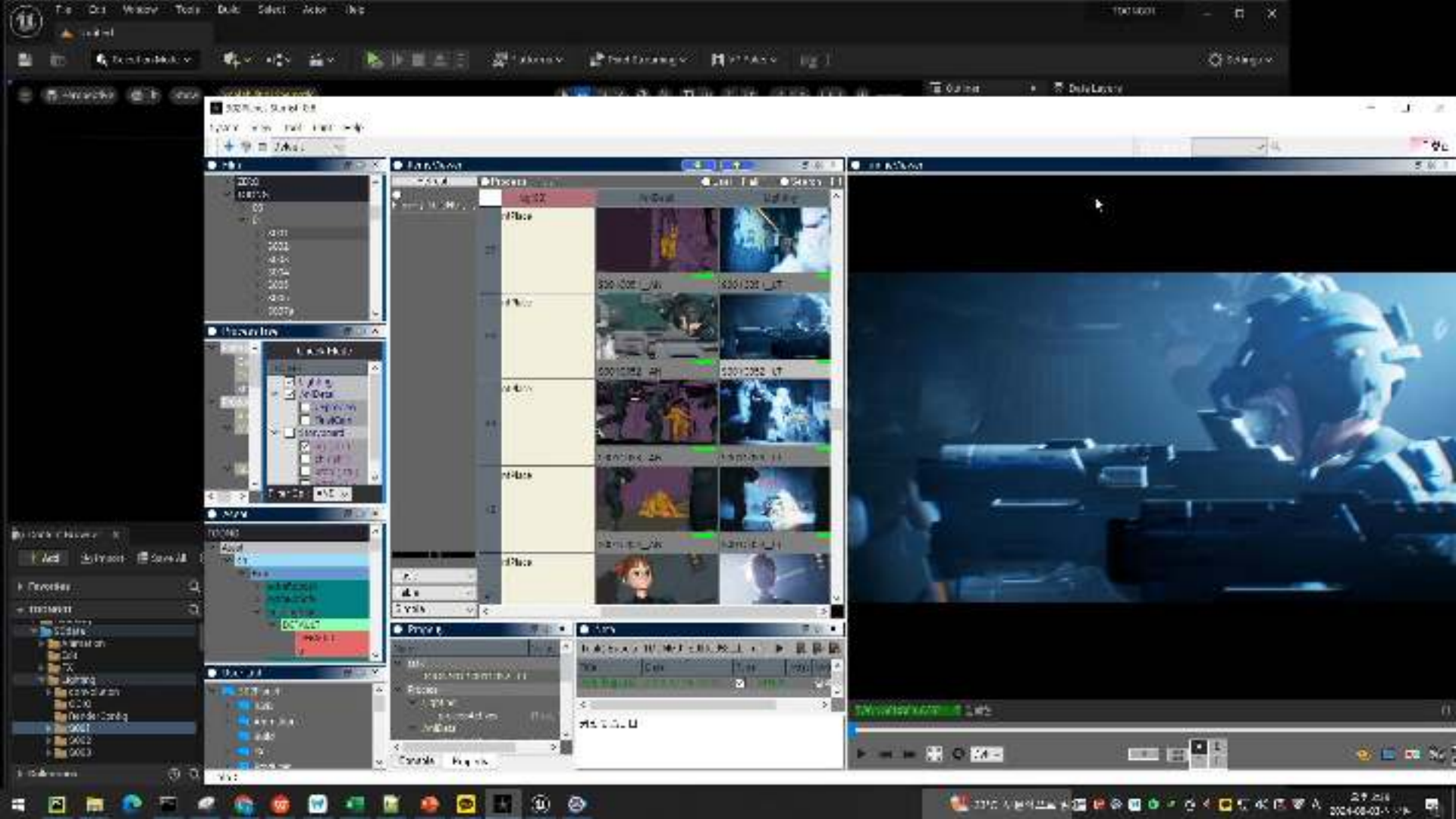
Create(Update)
Asset Blueprint

Open
Asset Blueprint

Create(Update)
Level Sequence

Open
Level Sequence

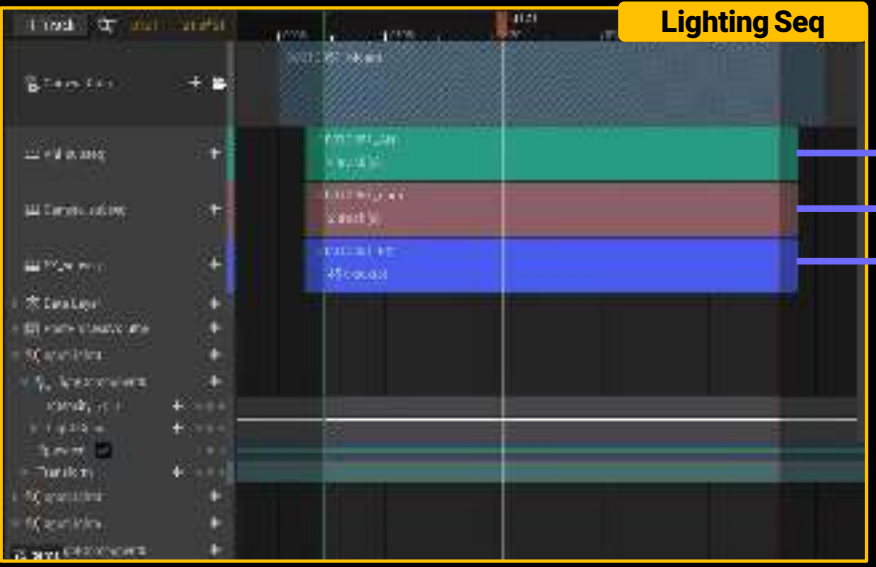
Add Job to
MRQ



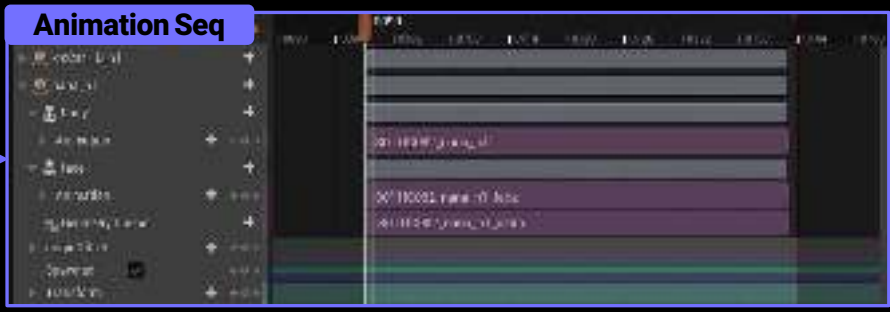
Sequence 구성

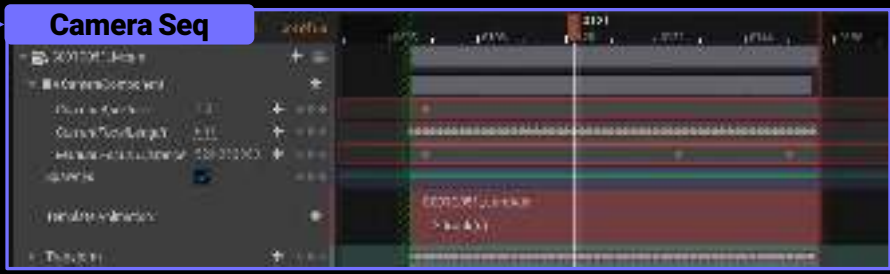
계층 구조를 이용한 Sequence 분산 관리

Main Sequence 계층 구조



SubSequence





각 공정별로 Sequence 생성 및 업데이트

공정별 작업 파일 충돌 없이 수정작업 동시 진행 가능

Camera Sequence 계층 구조

Camera Animation From Maya

Camera Seq

Hierarchical bias : 100

inherit

Template Animation

Modify focus, f-stop …
Add camera shake

Camera Modify Seq

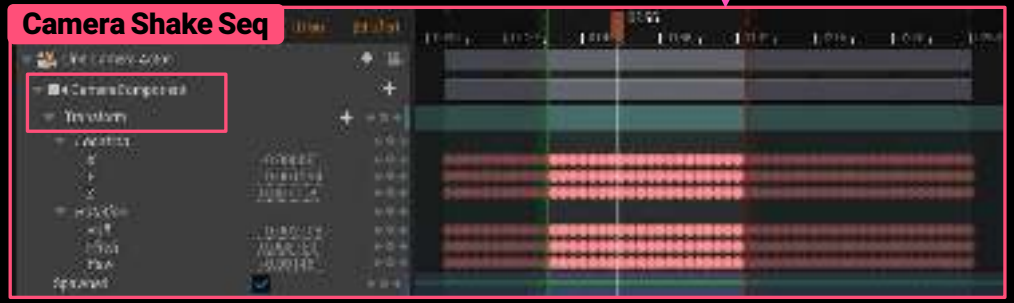
Hierarchical bias : 10

+ 100 = 110

Template Animation

Camera Shake Library

Camera Shake Seq



Template Animation 이용해서

Camera Animation을 override 하거나

Camera Animation을 추가한다

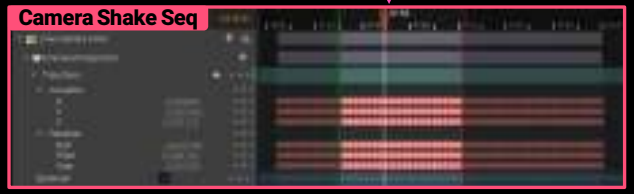
Sequence Priority

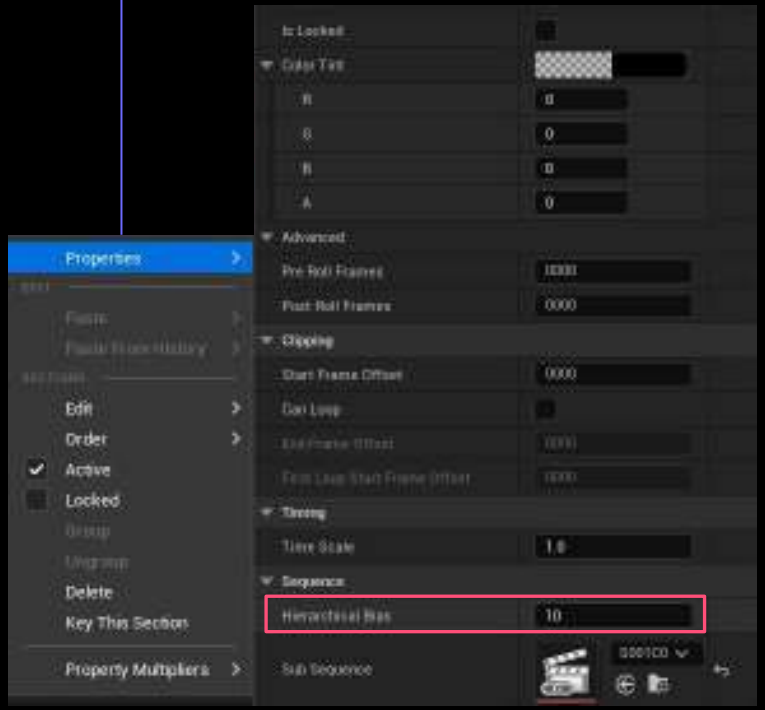
Hierarchical bias : 100

inherit

Hierarchical bias : 10

+ 100 = 110





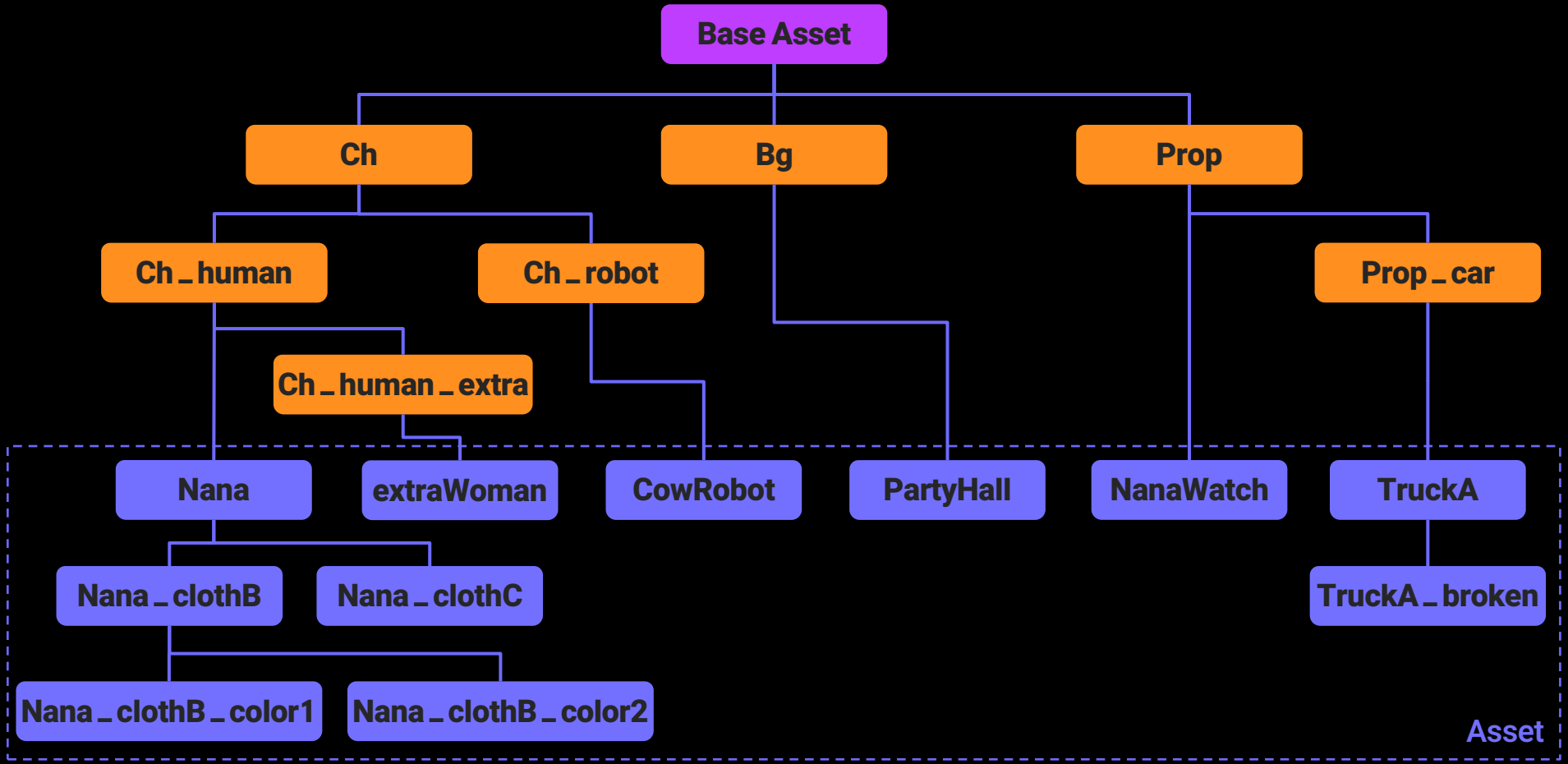
Hierarchical bias : + value → override

Hierarchical bias : 0 value → blend

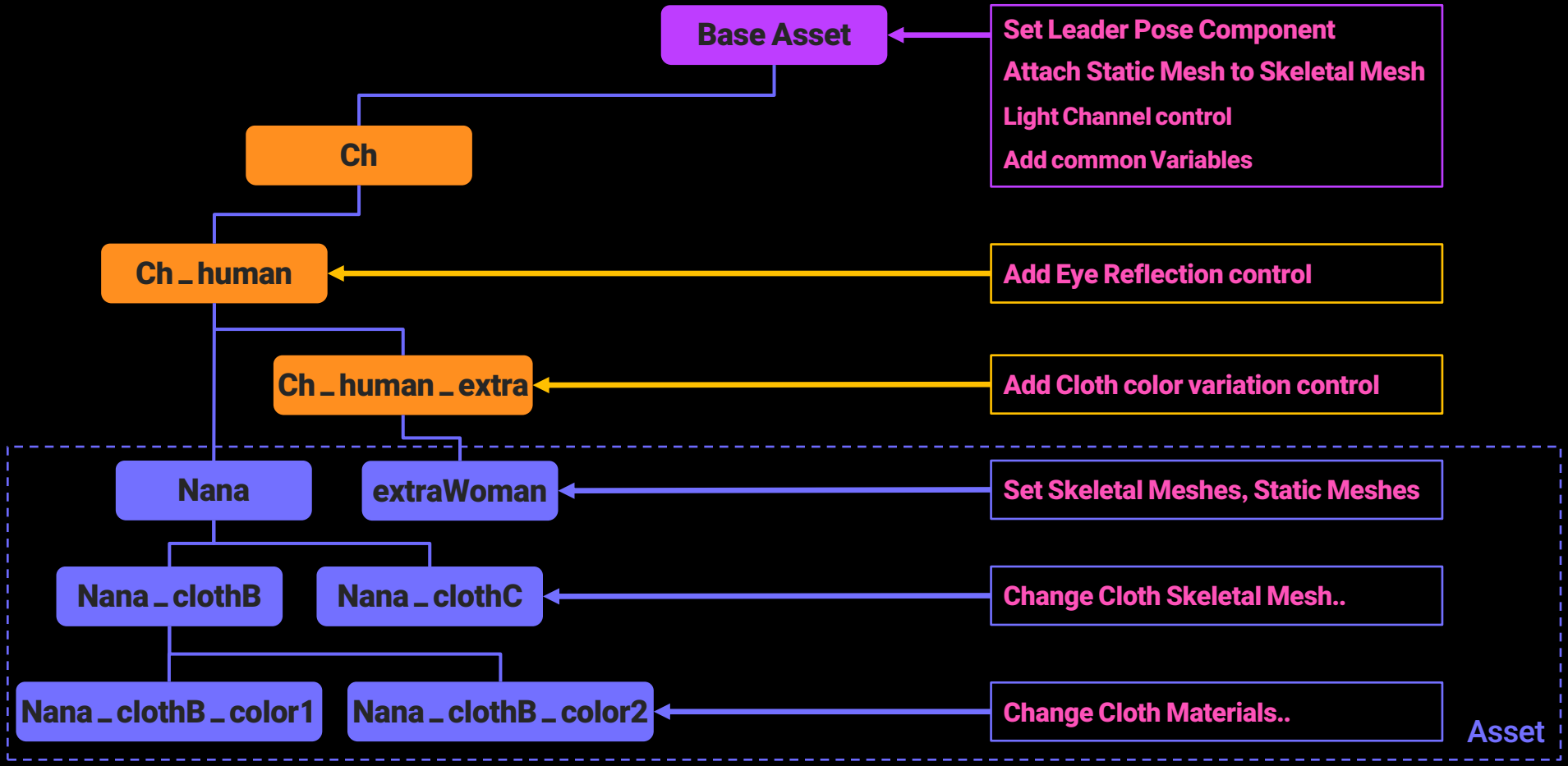
Asset Class 활용

Class 상속을 이용한 Blueprint Asset 관리

Asset Actor Class hierarchy



Asset Actor Class hierarchy



Section 3

애니메이션 제작 노하우 공유

- Hair 움직임 제어하기
- Eye Reflection 제어하기
- Instanced Static Mesh 이용하여 간단한 군중 만들기

Hair 움직임 제어하기

Guides Groom Cache를 이용하여 hair 움직임 제어하기


Fast / Slow motion

Unreal Engine의 hair simulation 사용시 문제점

Simulation bake(Caching)가 안된다.
Niagara Cache가 제대로 작동하지 않음.

MRQ로 렌더링할때 Simulation이 다르게 나온다.
Motion blur 렌더를 위해 Temporal Sampling을 적용하면 Sample 수 만큼 simulation시간이 늘어남.

추가적인 수정이 힘들다.
파고드는 오류 수정, 타이밍 조절, 연출상 필요한 헤어 키애니 작업 등이 힘들다.

다양한 Force 적용이 힘들다.
Turbulence, Vortex 등 다양한 외부 force들을 적용하기 힘들다

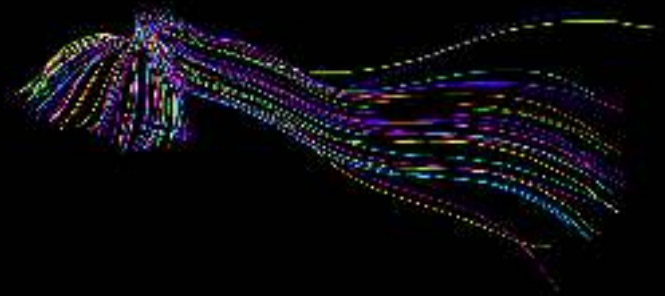
Hair Simulation Workflow

Maya
Simulation or Key Ani
Export Alembic

Strands Groom Cache

or

Guides Groom Cache

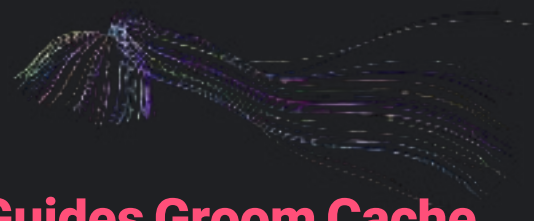


Unreal Engine
Import Alembic

Groom Cache Type 비교

Strands Groom Cache

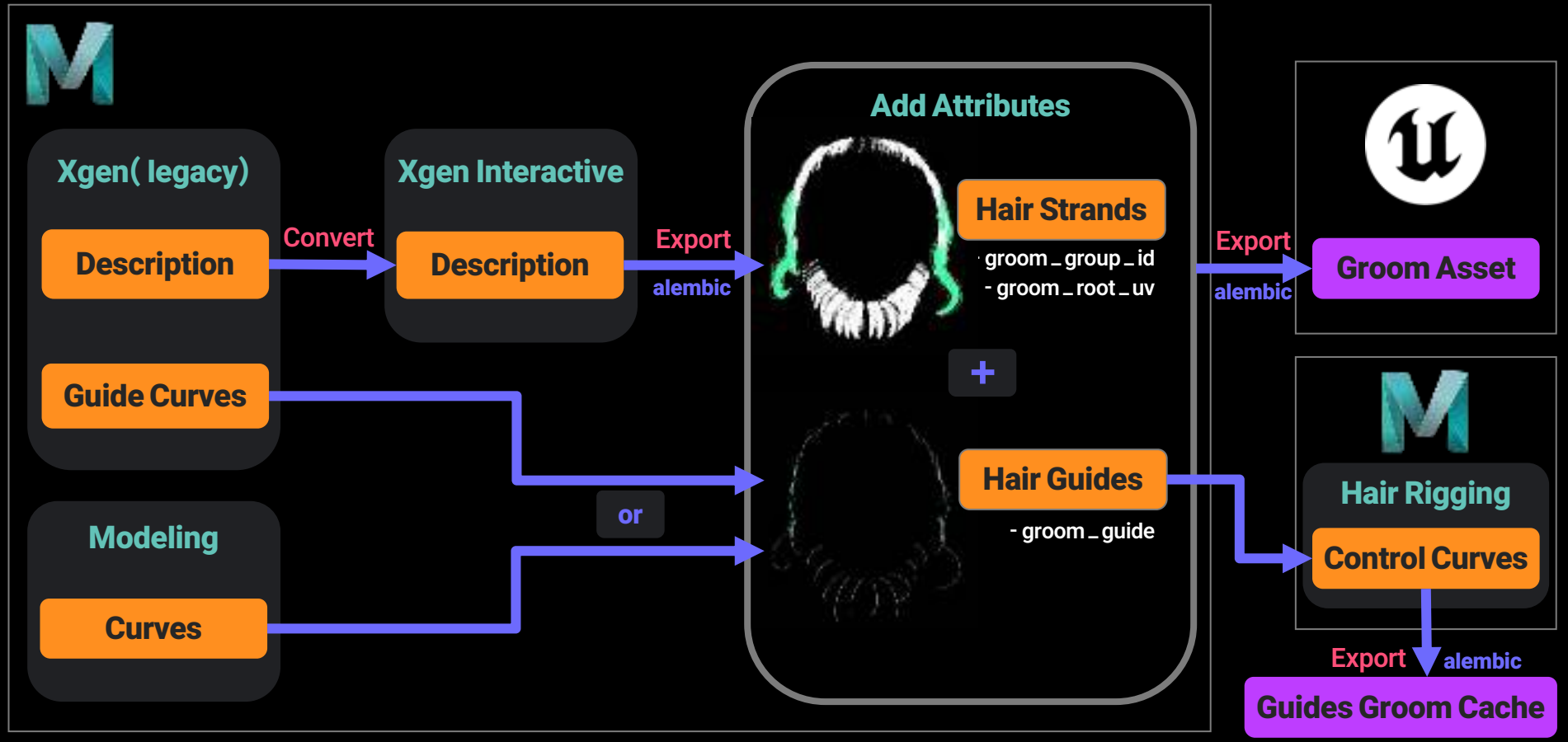
- Hair의 모든 Curve 데이타
- Vertex position, width, color 정보
- 무겁다 (memory ↗, performance ↘)
- UE Interpolation 사용하지 않음
- DCC의 simulation 모양 그대로 전달
- xgen에서 export 지원하지 않음



Guides Groom Cache

- Hair의 Guide Curve 데이타
- Vertex position 정보
- 가볍다 (memory ↘, performance ↗)
- UE Interpolation 사용
- Interpolation 방식에 따라 모양 달라짐
- Curve정보만 자유롭게 export

Guides Groom Cache를 사용할 수 있는 Groom Asset 만들기

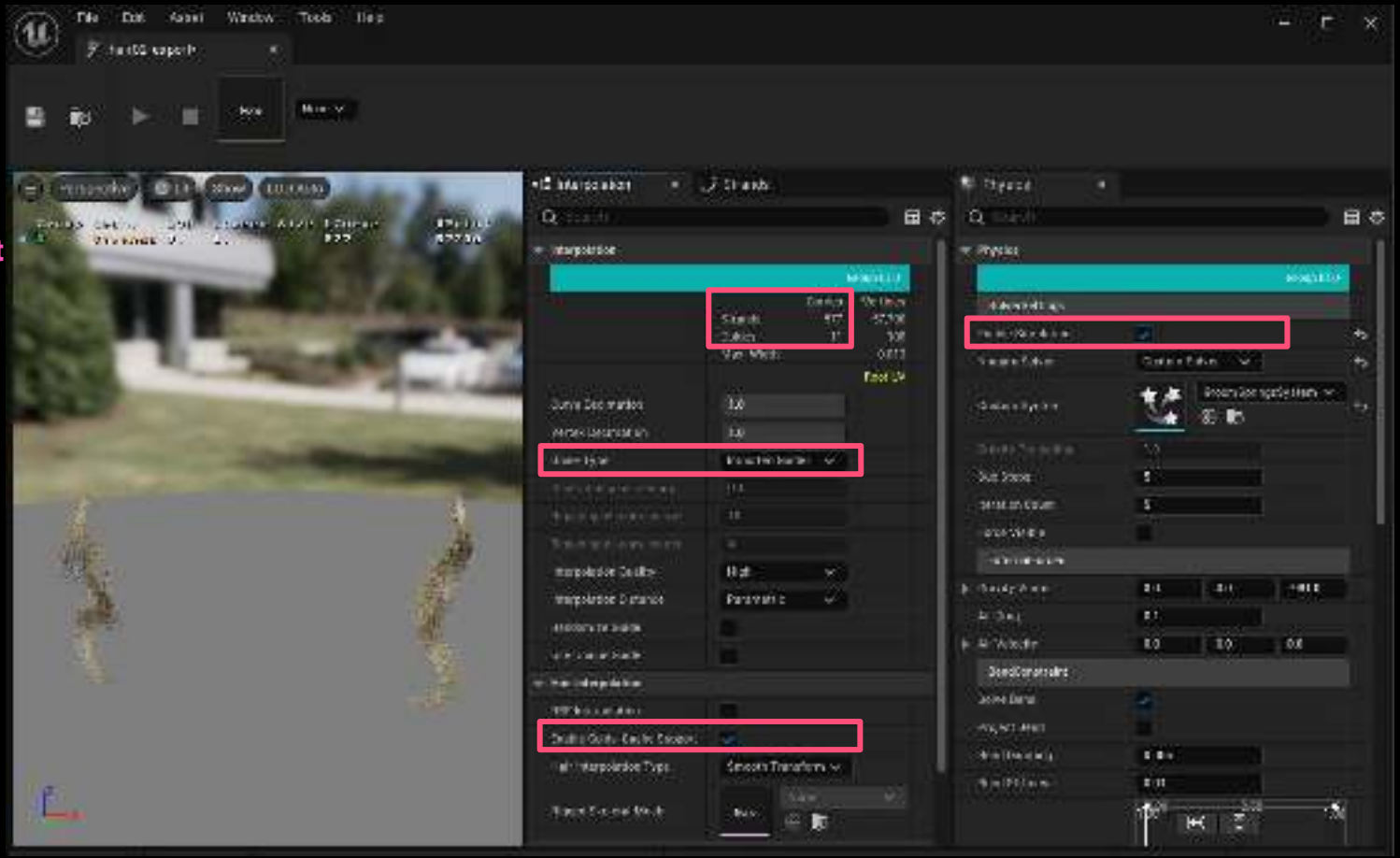


Groom Asset setting

Guide Type
- Imported Guides

Enable Guide-Cache Support
- On

Enable Simulation
- On



Groom Actor setting

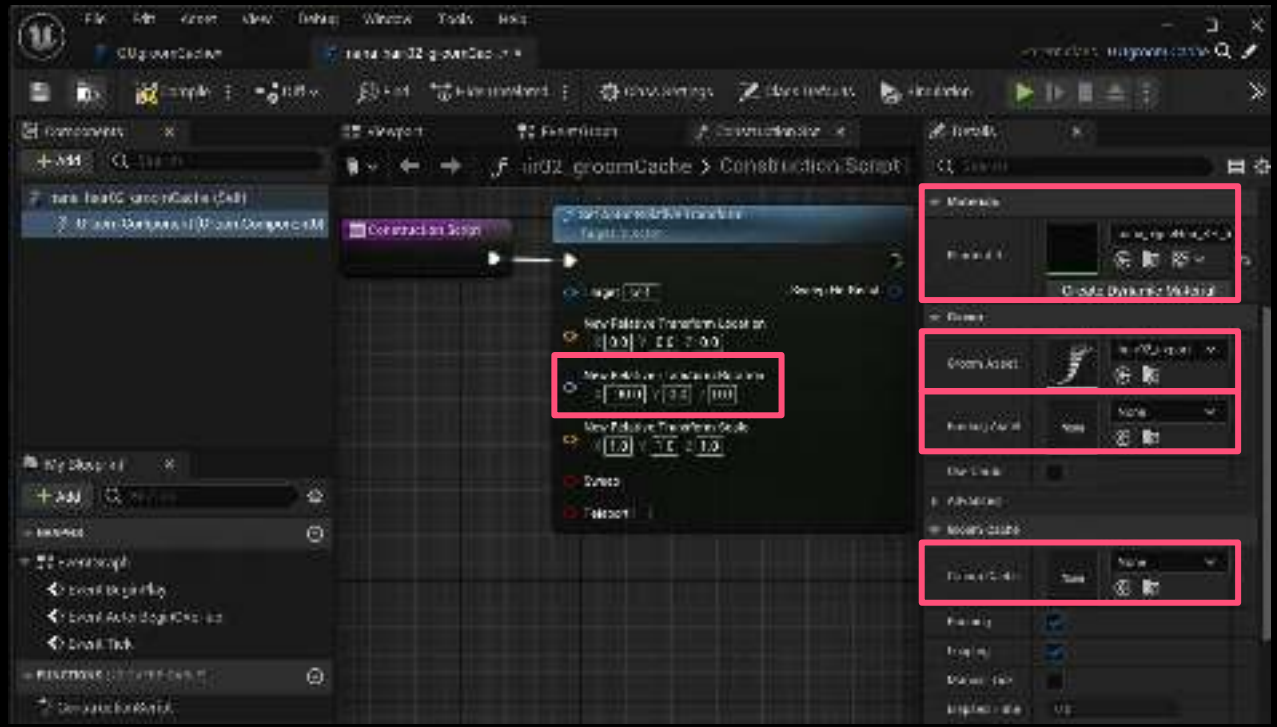
Create Groom Actor BP
- Character BP에 Groom Component로 포함시키면 시퀀서에서 Guides Groom Cache 에러 발생

Set Actor Relative Transform
- Construction Script
- Groom Actor를 Attach할 bone 방향에 맞추어 Rotation 조절

Set Groom Materials

Set Groom Asset

Set Groom Cache – None
- 시퀀서에서 추가

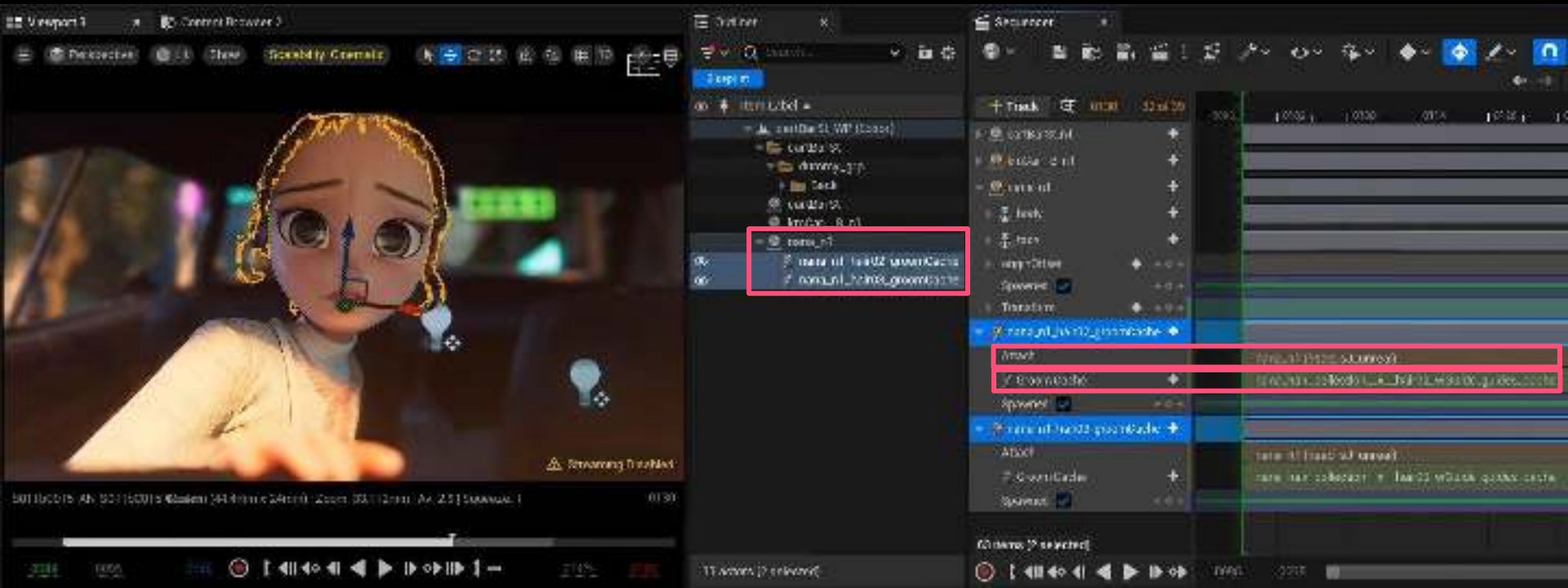


Sequencer에서 Guides Groom Cache 사용하기

시퀀서에 Groom Actor BP 추가

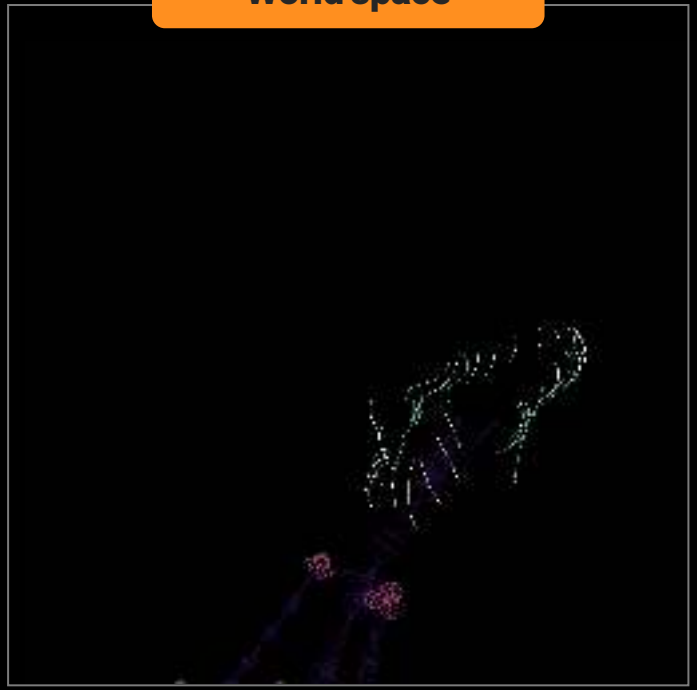
Guides Groom Cache 추가

캐릭터의 Head bone에 Attach



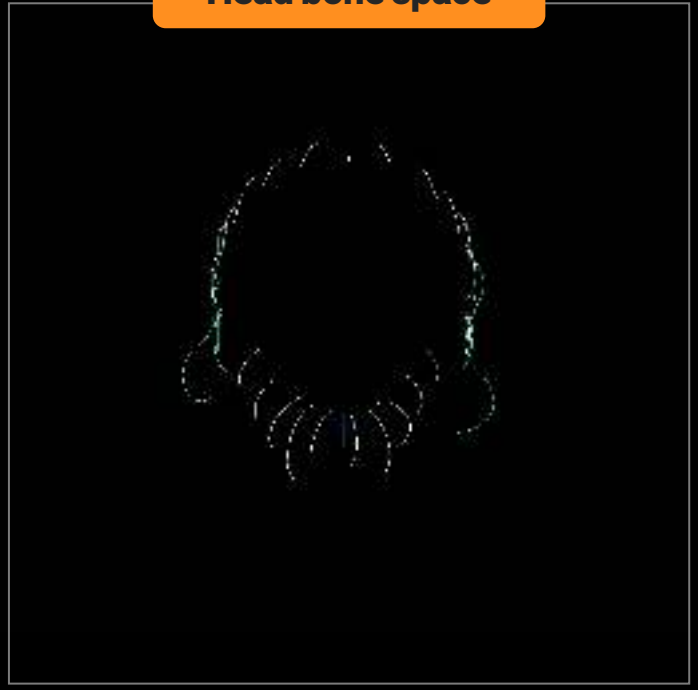
Guides Groom Cache from Maya

World space



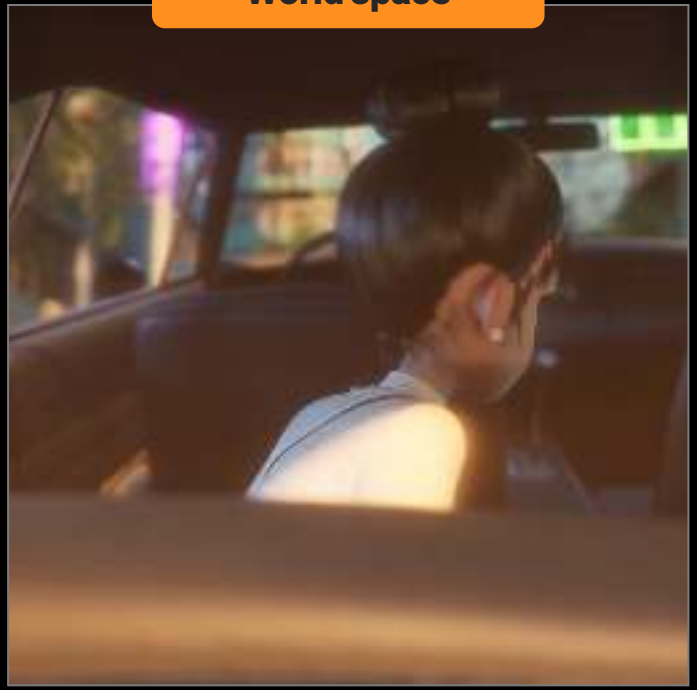
vs

Head bone space



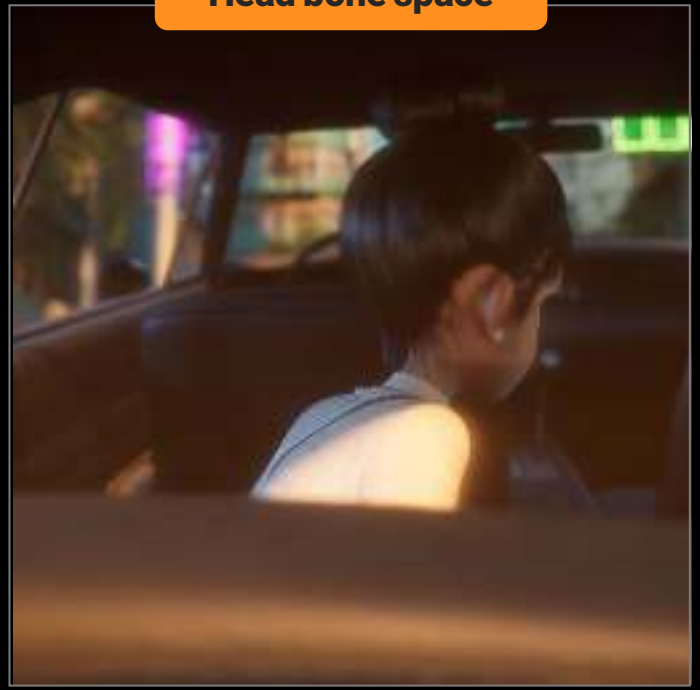
Guides Groom Cache Result

World space



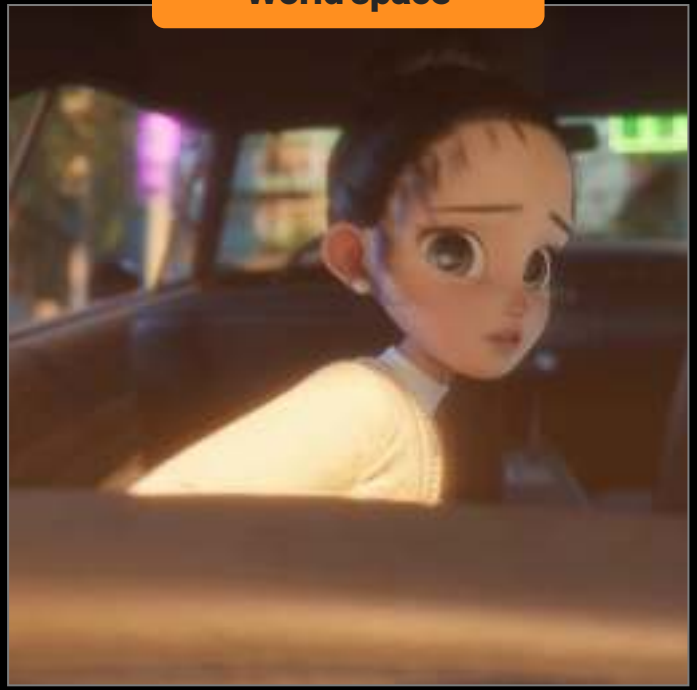
vs

Head bone space



Guides Groom Cache Result

World space

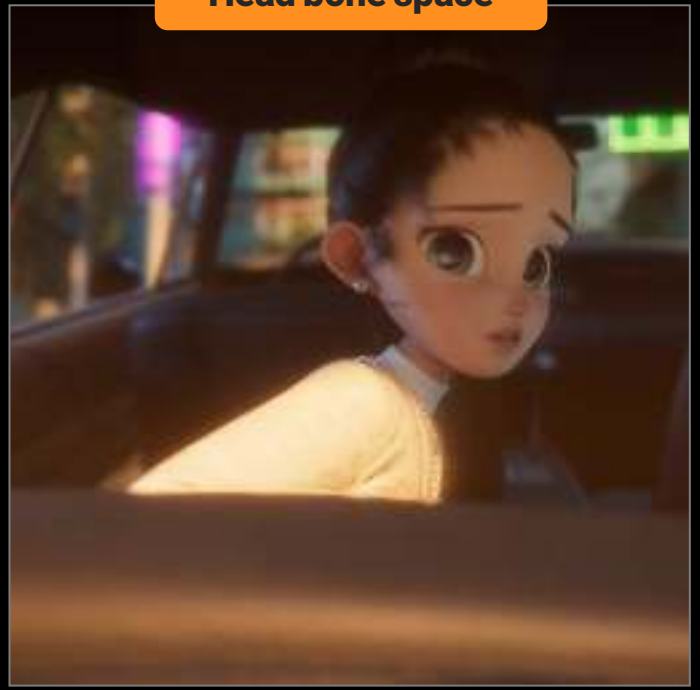


Head rotation시 문제 발생

- Groom이 Head에 binding되는 정보 없이 Guide Curve만으로 Rotation 한계

vs

Head bone space



Head Attach로 문제 해결

- Head bone에서 Rotation정보를 가져오고 Guide Curve는 Position정보만 업데이트

Eye Reflection 제어하기

Sequencer에서 Eye Reflection 실시간 제어하기 (Fake / Real)

Real Eye Reflection 사용시 문제점

주변 배경 및 조명 상황에 따라 정해진다.

- 원하지 않는 모양 및 위치에 Reflection이 만들어 질 수 있다.
- Eye Reflection만을 위한 추가 조명 설치가 필요하다.

Fake Eye Reflection 요구 사항

상황에 맞는 여러 가지 Reflection Map 이용 가능

Reflection Map의 위치 및 크기 조절 가능

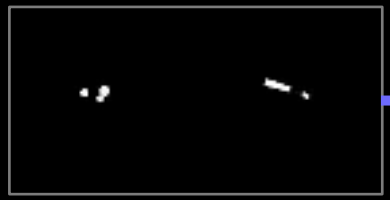
캐릭터 Blueprint를 통해 직관적으로 제어

Sequencer에서 확인하며 실시간으로 제어
- 위의 모든 요구 사항을 Sequencer에서 조절 가능

Eye Reflection Map types

Highlight

Highlight Image
(from Library)

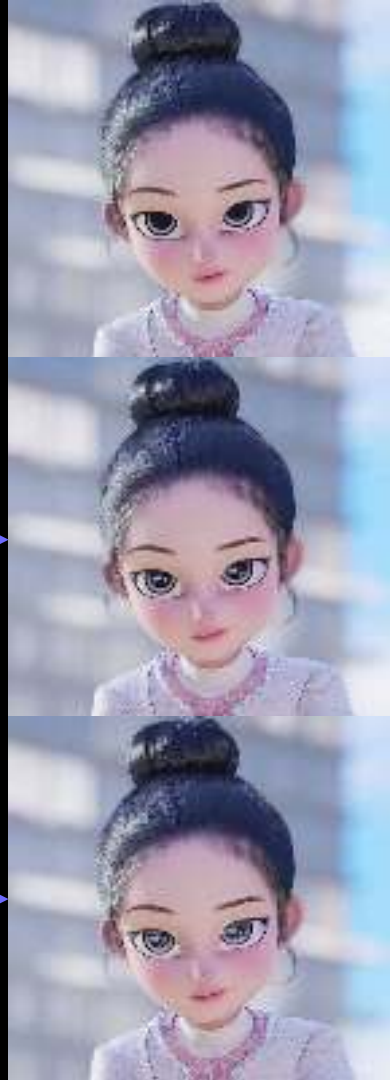


Environment

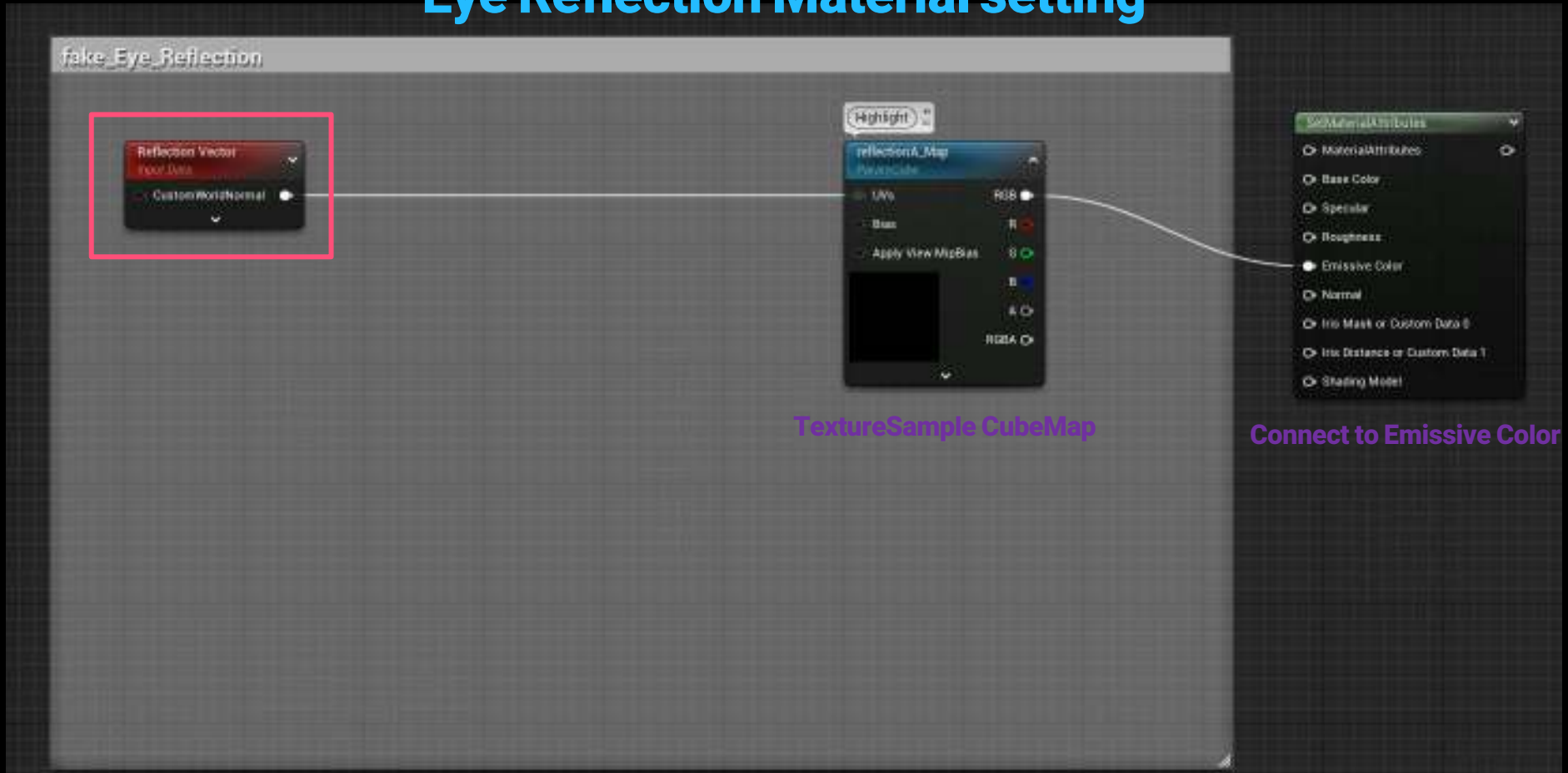
Environment Image
(from Library)

Current Scene Capture
(One Frame)

Current Scene Capture
(Every Frame)



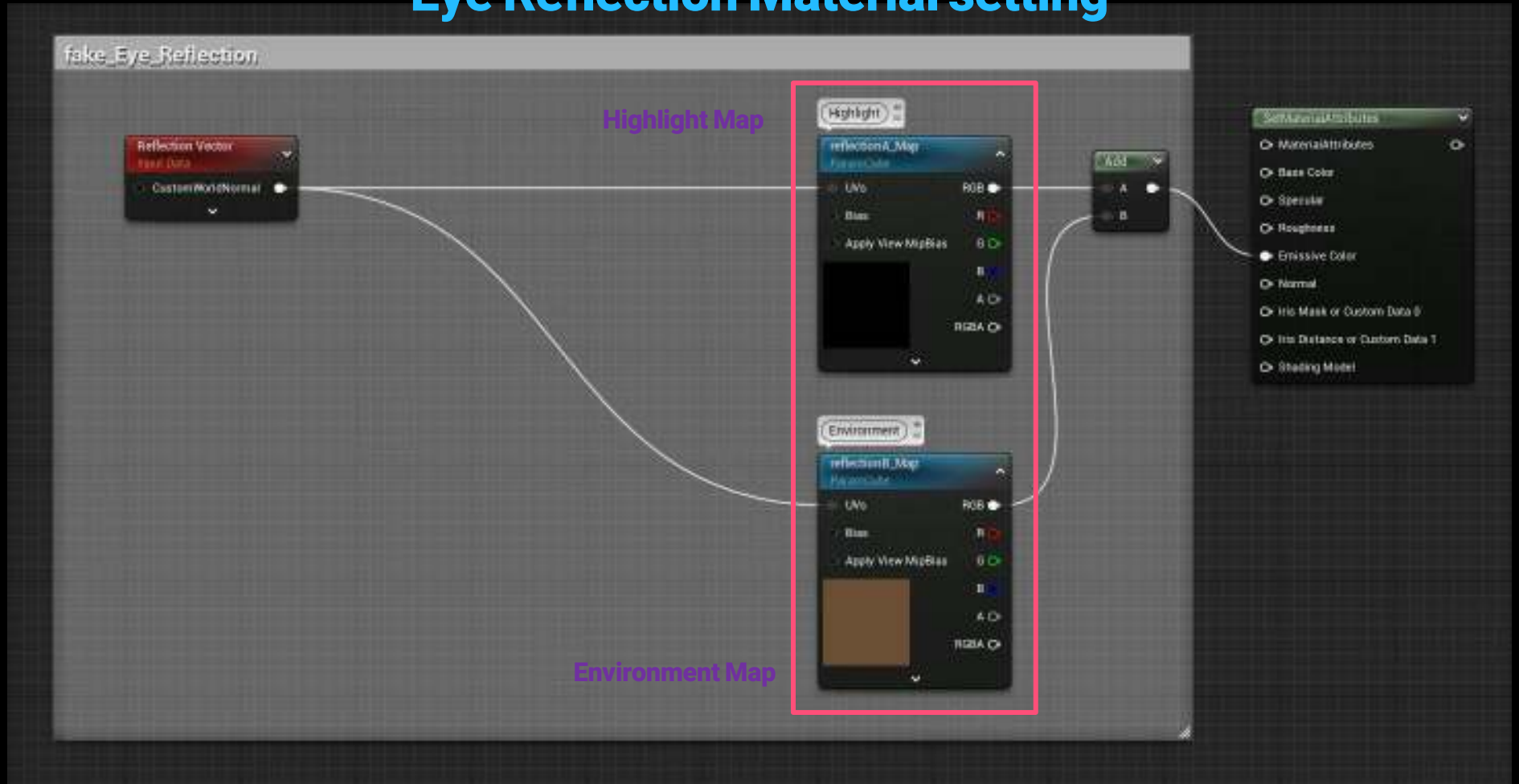
Eye Reflection Material setting



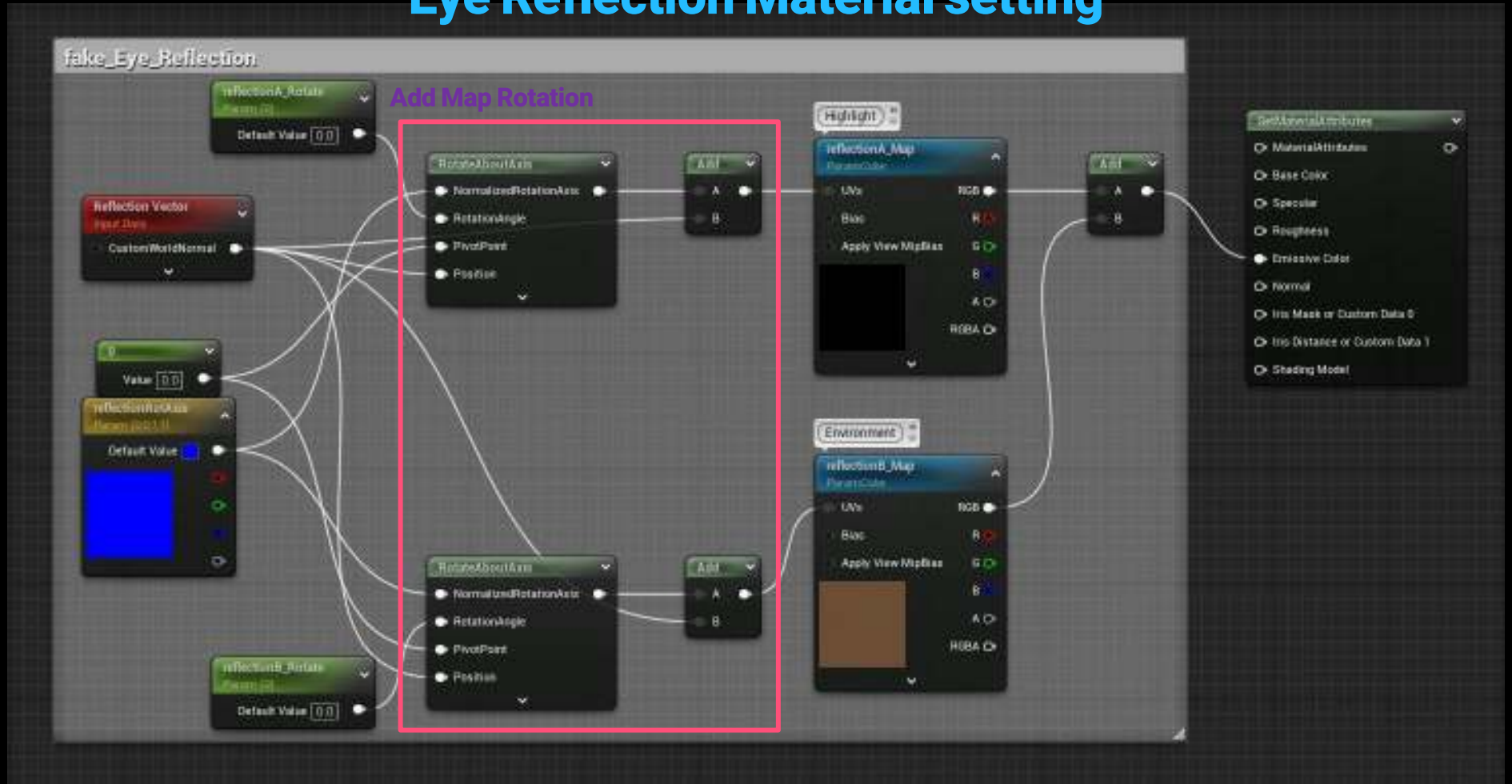
TextureSample CubeMap

Connect to Emissive Color

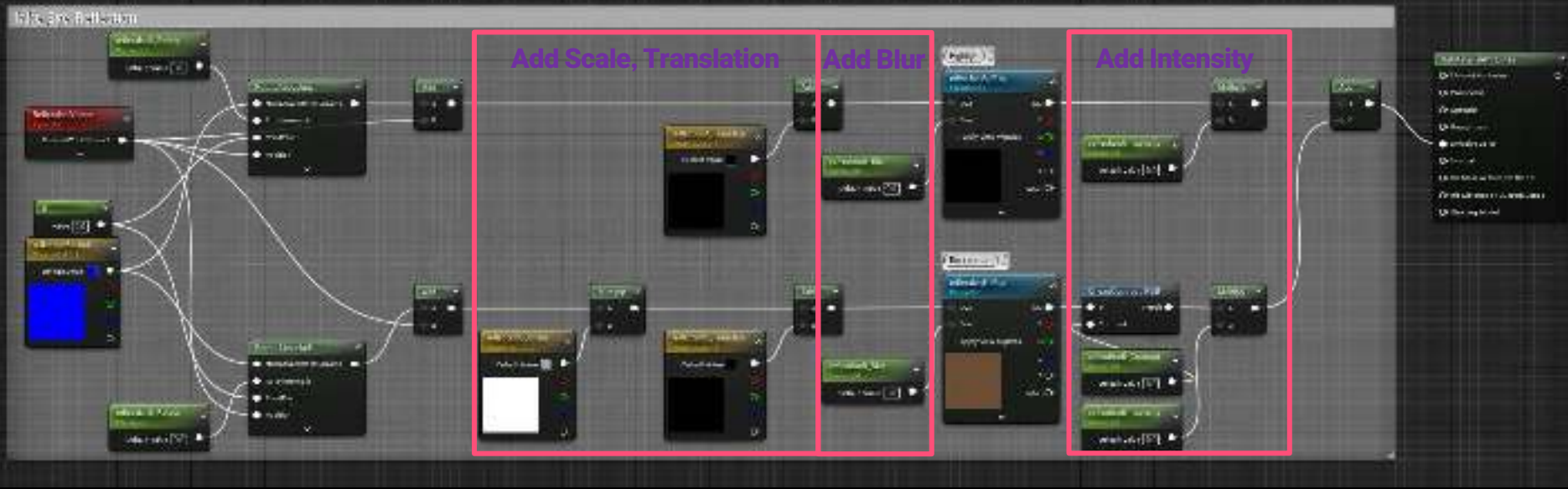
Eye Reflection Material setting



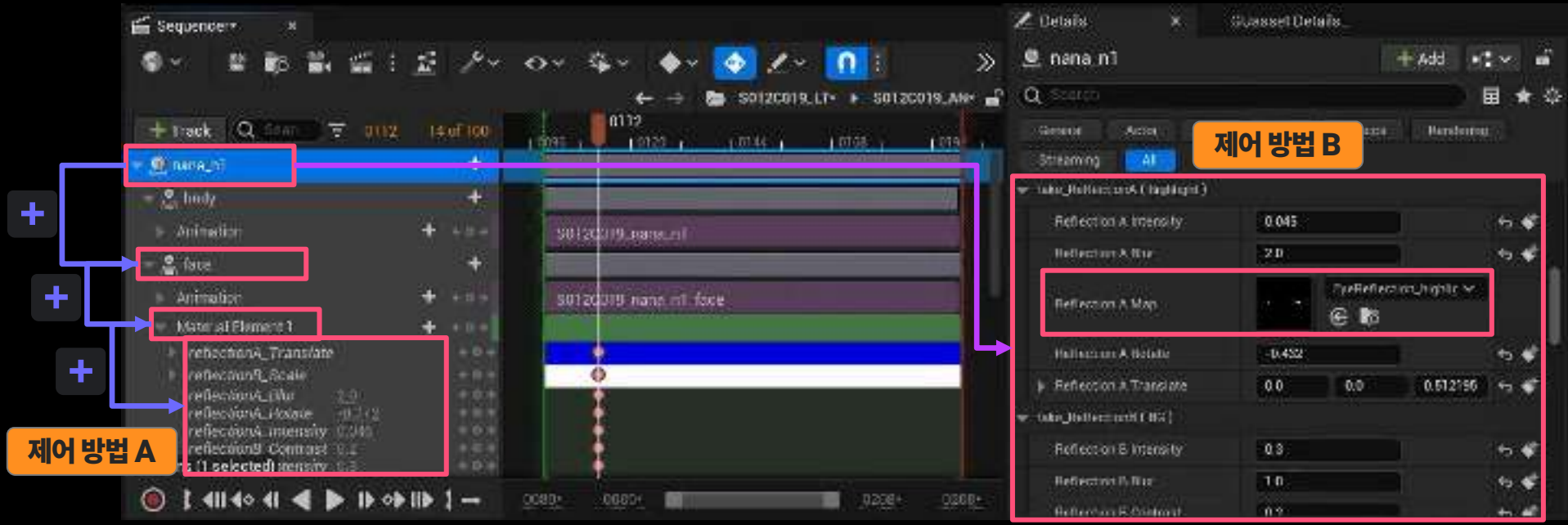
Eye Reflection Material setting



Eye Reflection Material setting



Sequencer에서 Eye Reflection Material 제어하기



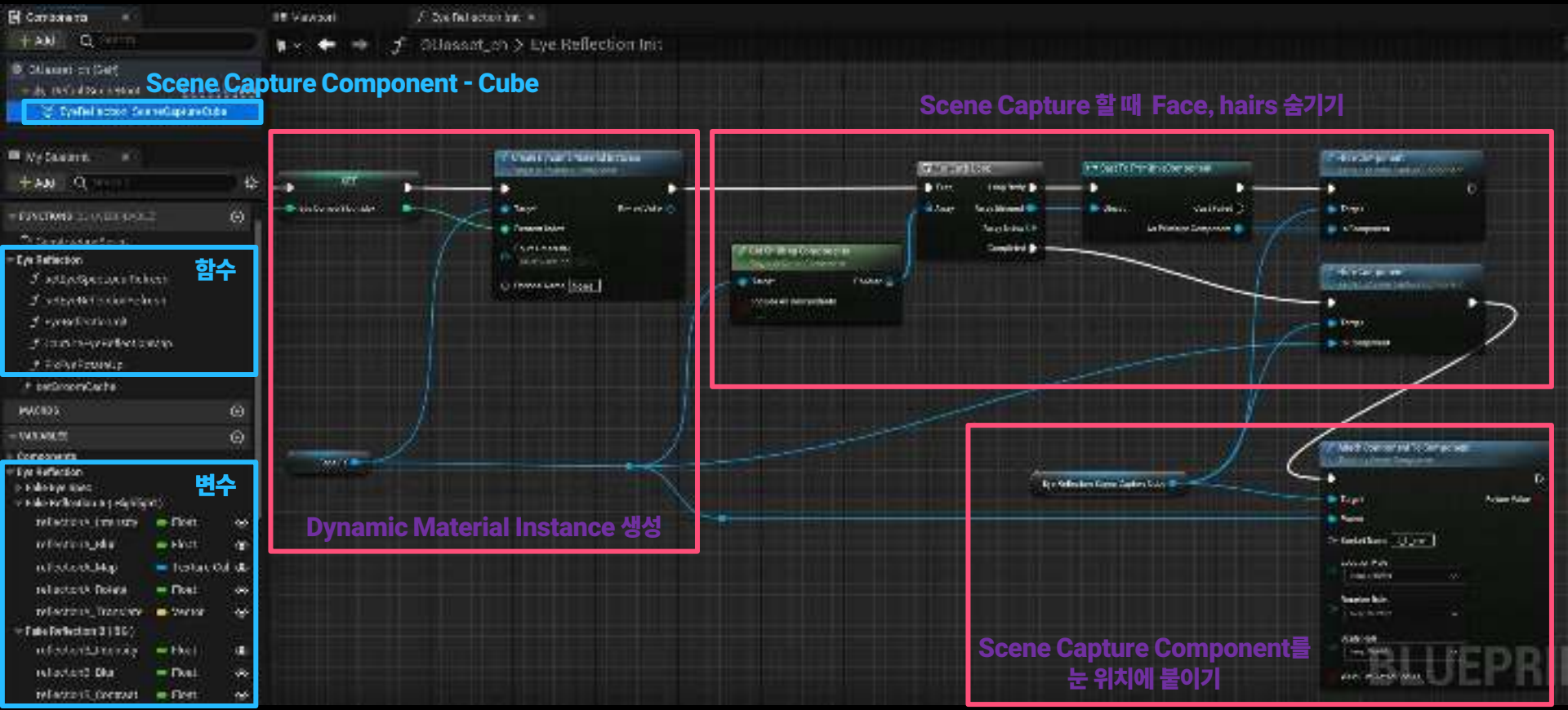
제어 방법 A

- Character BP ->skeletal Mesh ->Eye Material ->Reflection 변수 추가 및 제어
- Reflection Map 변경 불가능
- Character BP 추가 셋팅 필요 없음

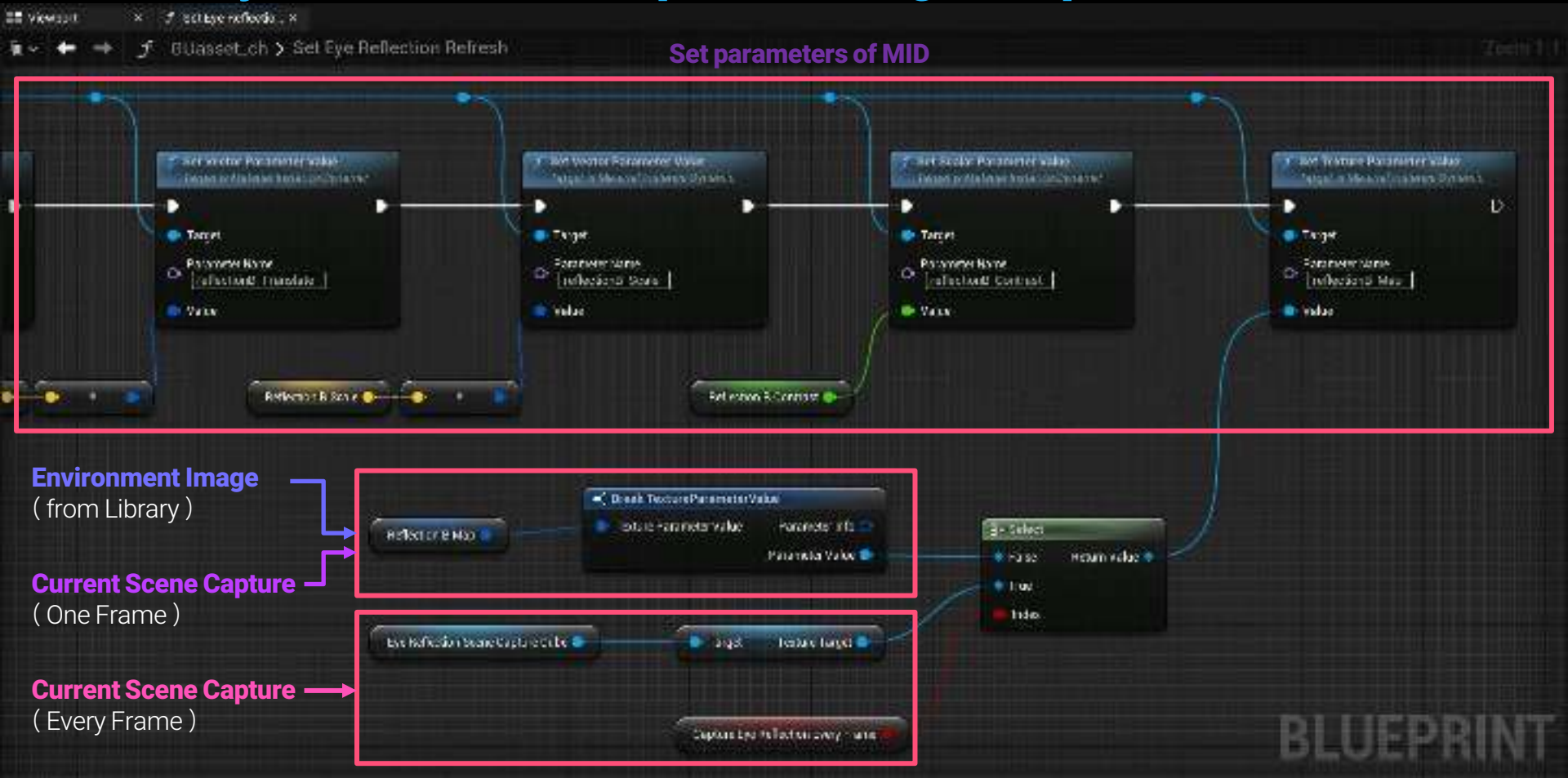
제어 방법 B

- Character BP ->Reflection 변수 제어
- Reflection Map 변경 가능
- Character BP 추가 셋팅 필요

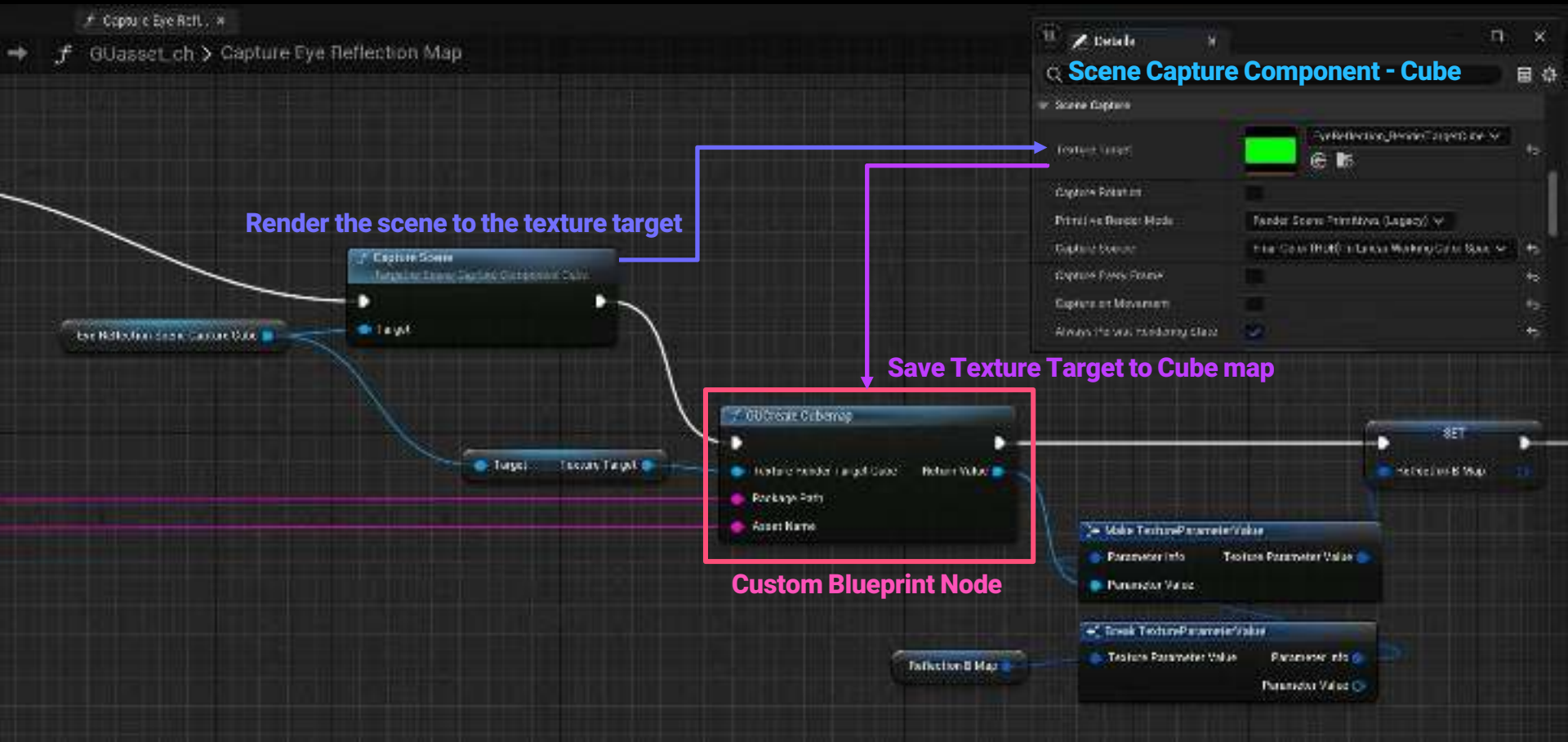
Eye Reflection Blueprint setting (init)



Eye Reflection Blueprint setting (set parameters)



Eye Reflection Blueprint setting (Scene Capture)



Custom Blueprint Node (Create Cube Map)

Blueprint Node Texture Target에 Render된 이미지를 지정한 경로의 지정한 파일 이름으로 저장한다.

C++ Code

```
UTextureCube* UGUfunctionLibraryBPLibrary::GUCreateCubemap(class UTextureRenderTargetCube* TextureRenderTargetCube,
                                                           FString PackagePath, FString AssetName)
{
#if WITH_EDITOR
    UPackage* mPackage = CreatePackage(*PackagePath);

    UTextureCube* mTextureCube = TextureRenderTargetCube->ConstructTextureCube(mPackage, AssetName,
                                                                               EObjectFlags::RF_Public|EObjectFlags::RF_Standalone);

    if (mTextureCube)
    {
        mTextureCube->MarkPackageDirty();
        mTextureCube->GetOuter()->MarkPackageDirty();
        FAssetRegistryModule::AssetCreated(mTextureCube);
    }
    return mTextureCube;
#else
    return nullptr;
#endif
}
```

Input : Texture Target, Path, FileName

Main Process : Create Cube Map

Output : Cube Map

Sequencer에서 Character BP 변수 제어할 때 문제점 및 해결

BP의 Details view에서
변수 제어하면
Construction Script
실행되면서
초기 pose로 돌아감

➡ Viewport를 보며
실시간 제어가 힘들다

↓

Custom Detail View
사용하여 문제 해결

Custom Details View (Editor Utility Widget)

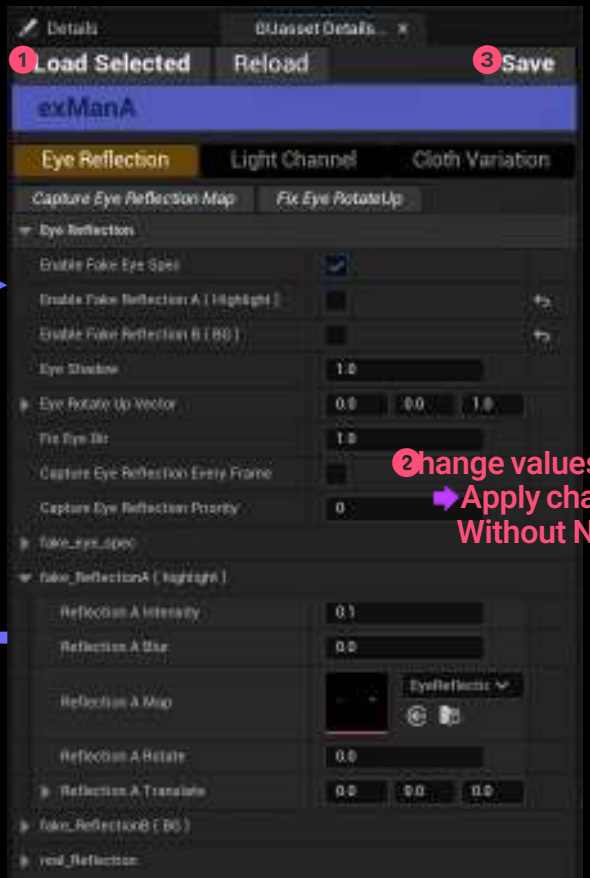
Character Blueprint

기존의 Details View의 변수들을
원하는 대로 구조화
원하는 대로 제어

Editor Utility Widget

1 et current values

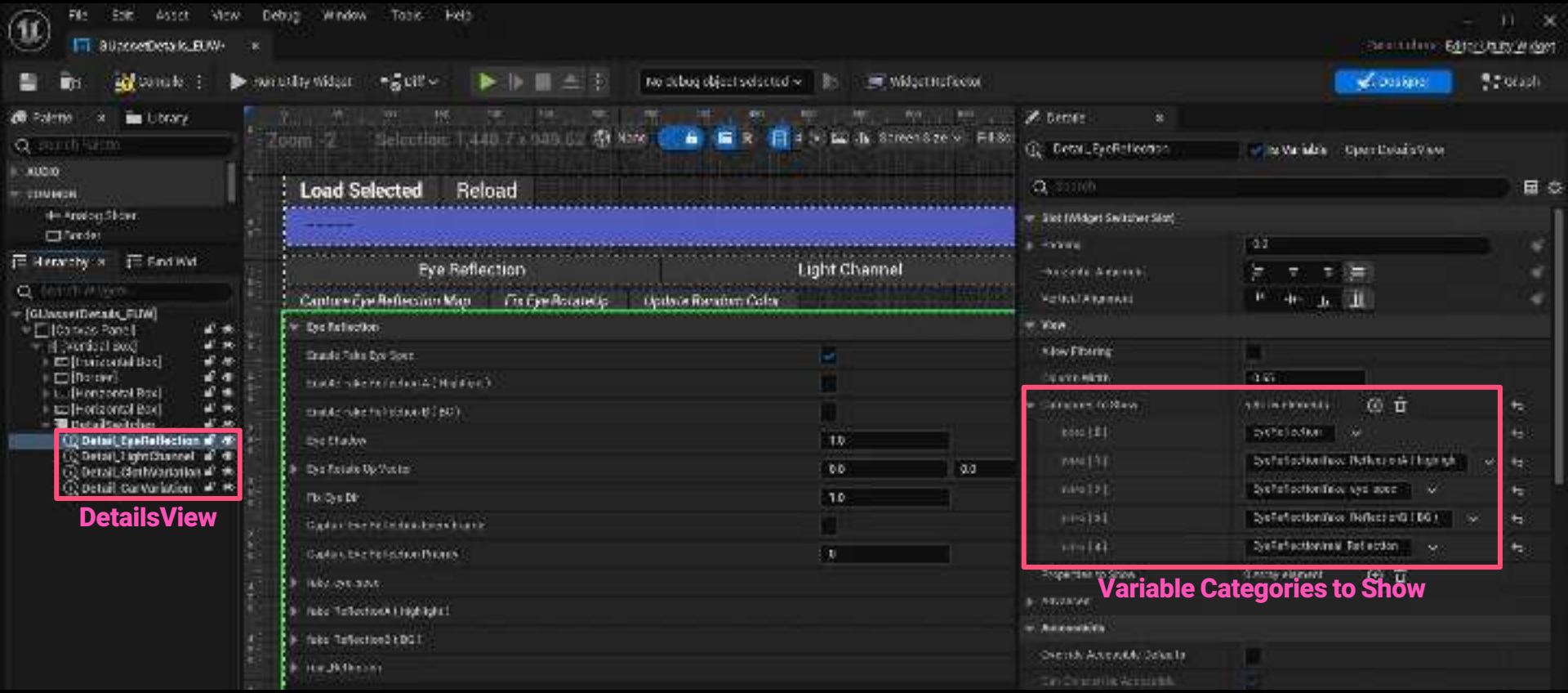
3 ave changed values



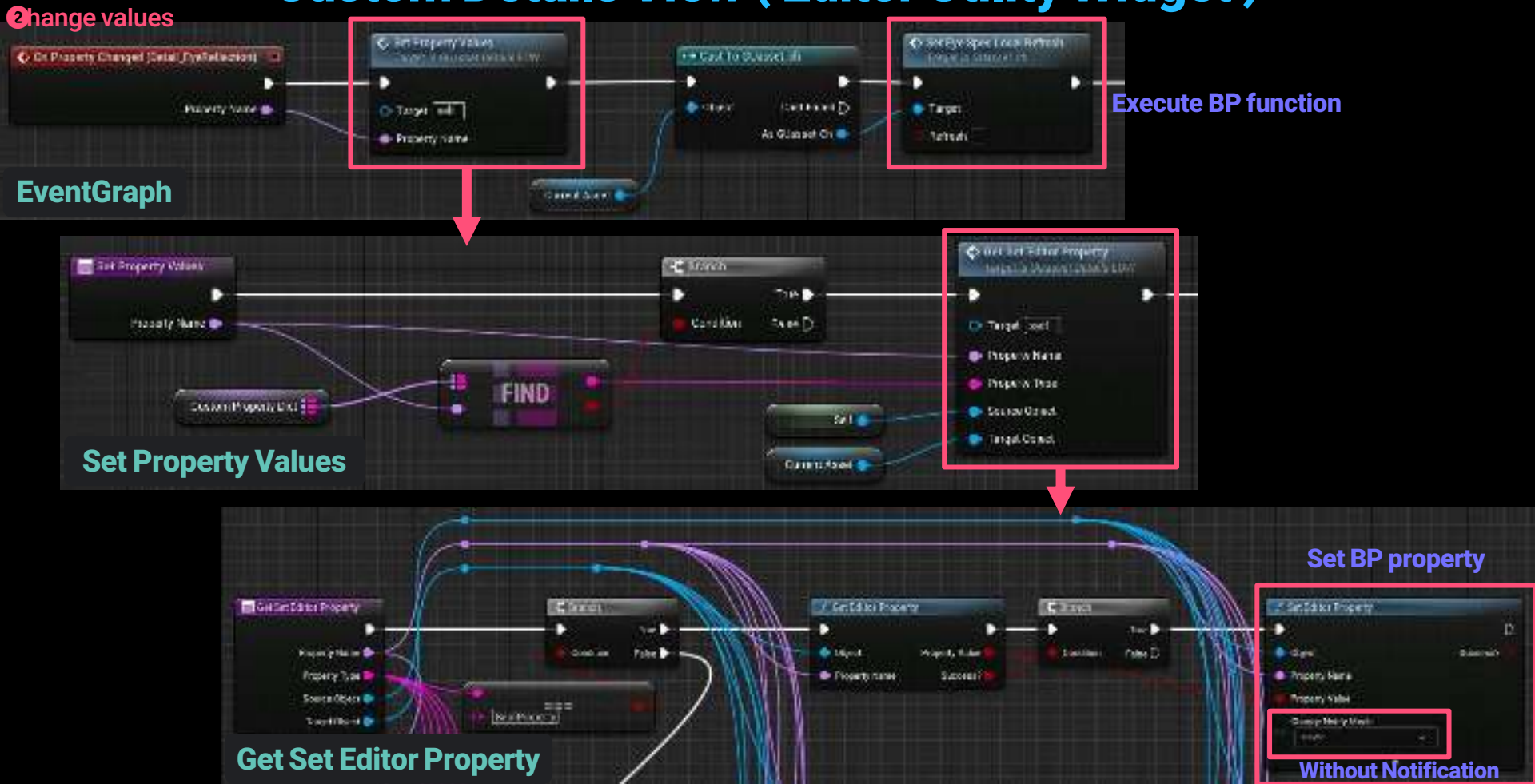
2 hange values
Apply changed
Without Notification

Custom Details View (Editor Utility Widget)

Editor Utility Widget 을 이용하여 캐릭터 Blueprint의 변수들을 제어한다



Custom Details View (Editor Utility Widget)



Instanced Static Mesh 이용하여 간단한 군중 만들기

날아다니는 Drone들 배치 및 Robot관 배치


11.18.Wed
PM 13:36:21

Instansed Static Mesh(ISM) 의 특징

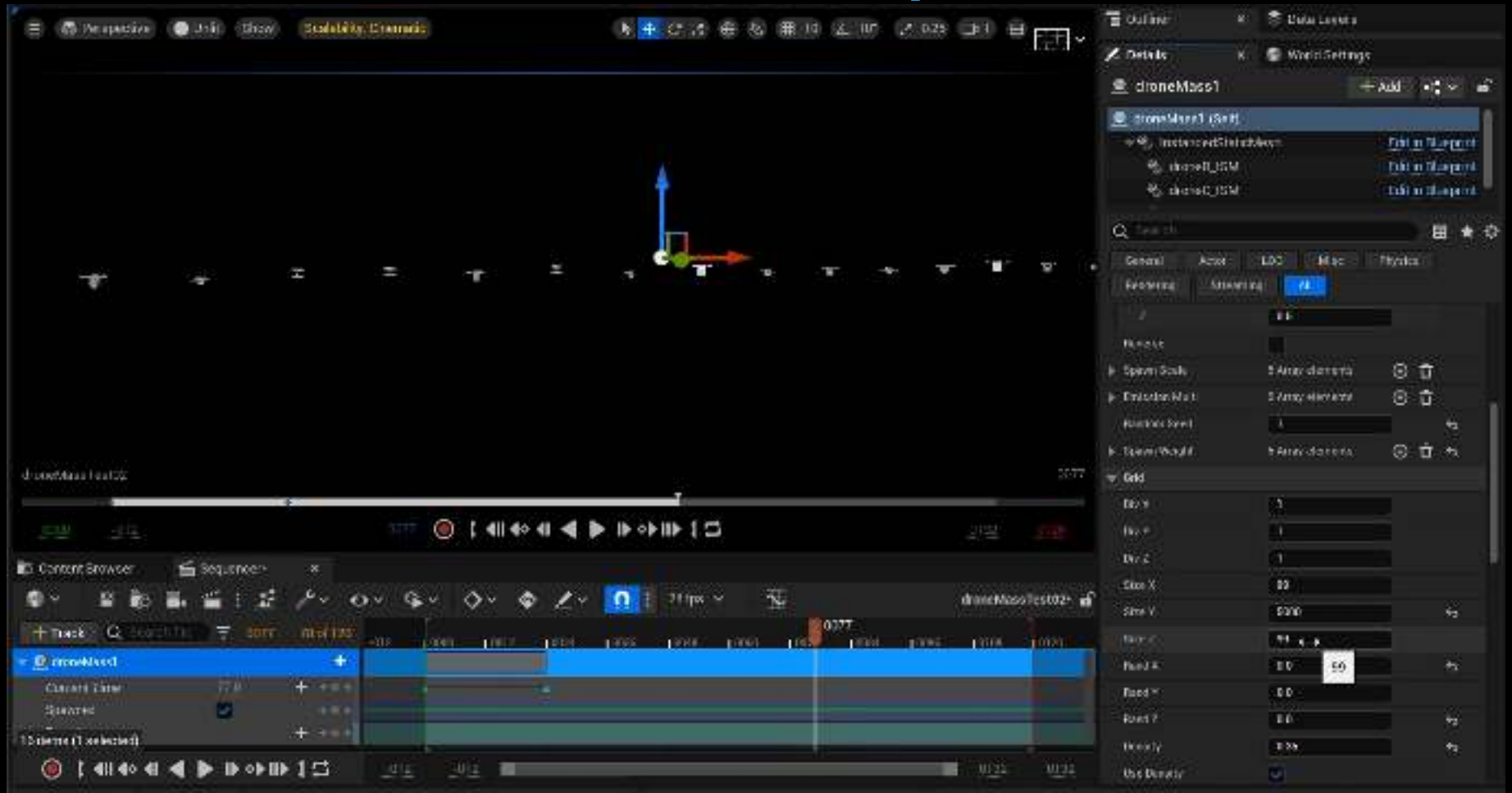
동일한 Static Mesh가 반복될 경우 사용

하나의 Static Mesh 정보를 가지고 수많은 Instance(copy) 표현

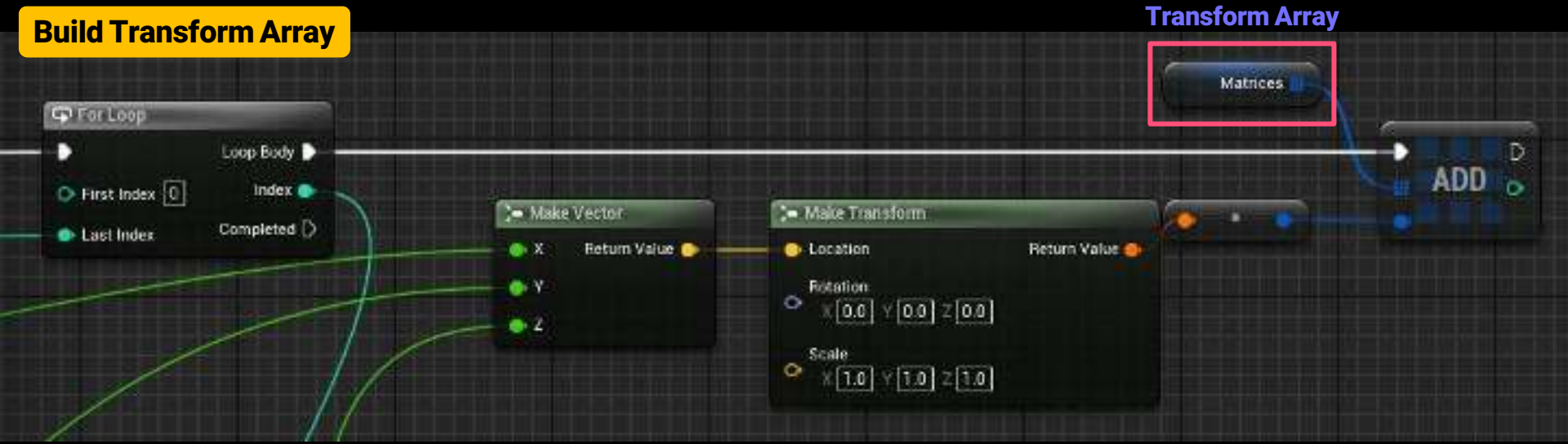
가볍다 (memory ↘, performance ↗)

수많은 Instance들의 Transform, Custom Data 정보를 BP에서 일괄 제어 가능

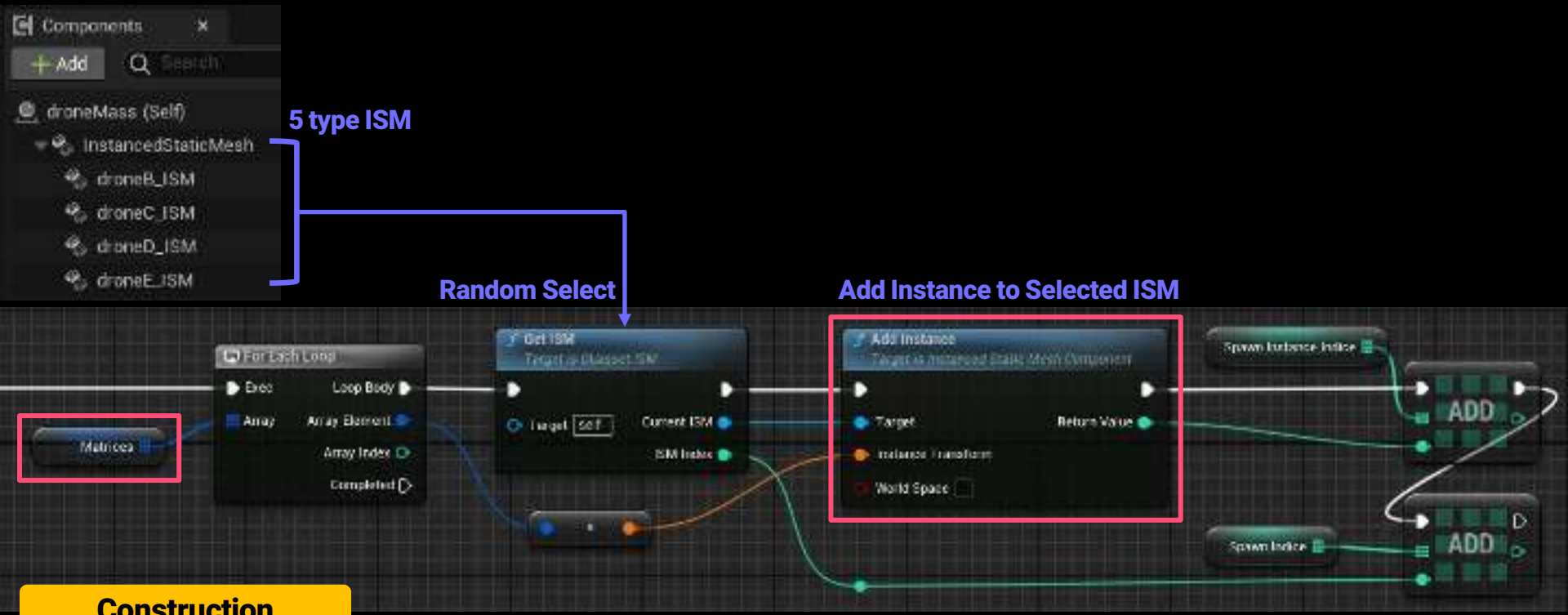
Drone Crowd(ISM blueprint) 제어



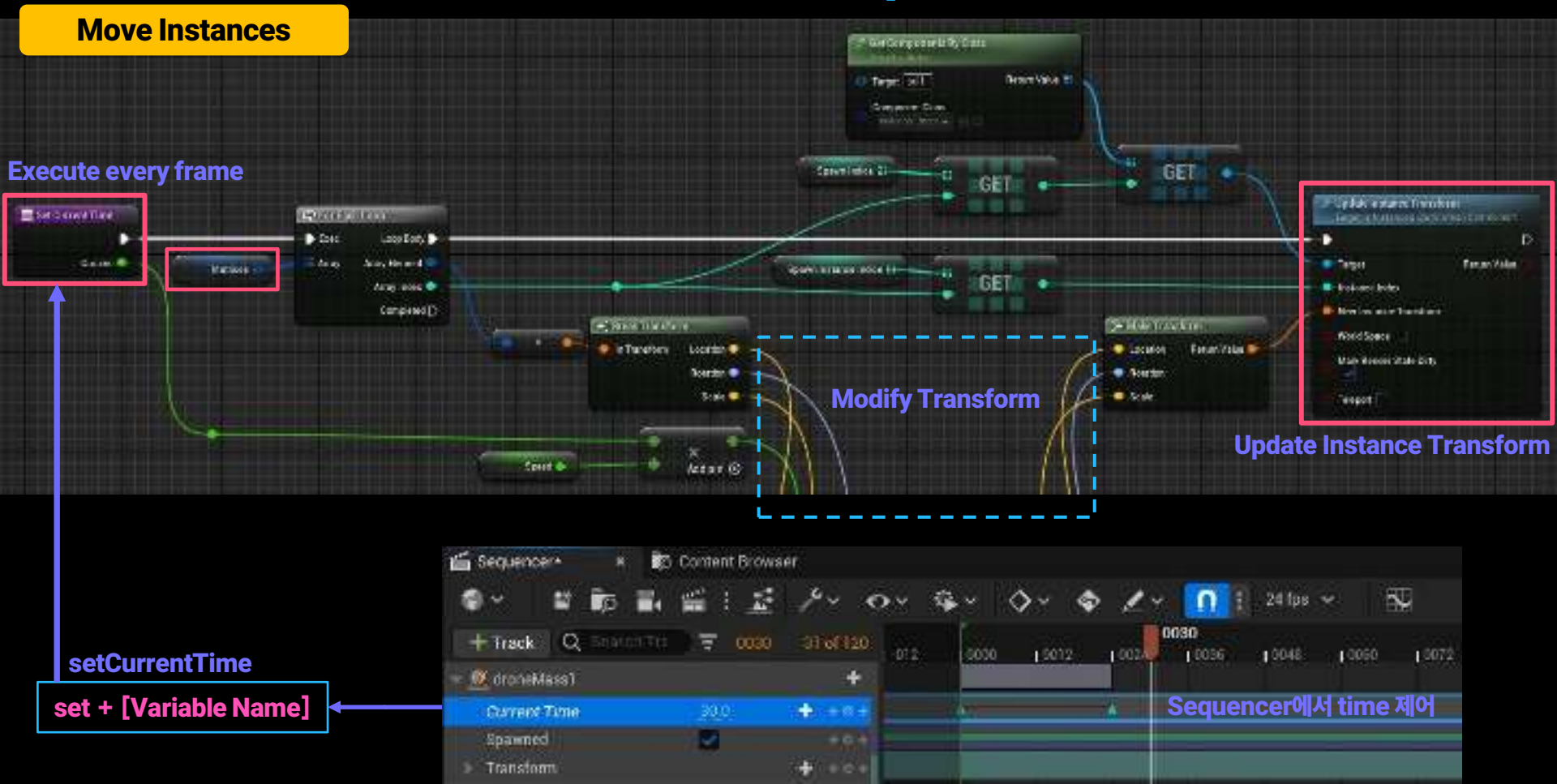
Drone Crowd(ISM blueprint) 작동 구조



Drone Crowd(ISM blueprint) 작동 구조



Drone Crowd(ISM blueprint) 작동 구조



Robot Crowd(ISM blueprint)

Static Mesh

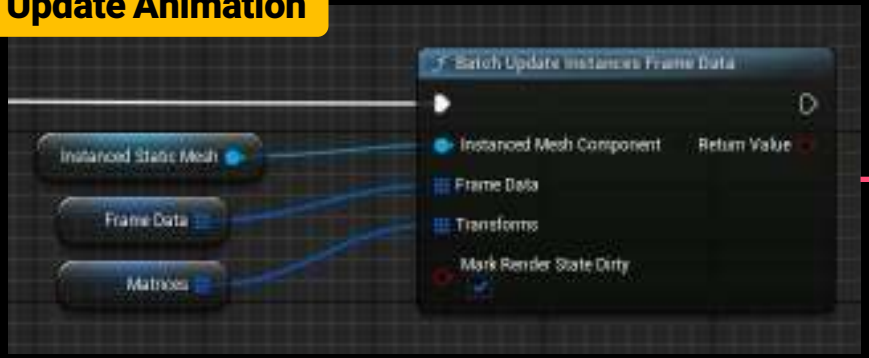
AnimToTexture plugin 사용

Skeletal Mesh의 애니메이션 정보를 Texture로 bake하여
Static Mesh의 머티리얼에 적용
(World position offset에 적용)

Baked Textures

Robot Crowd(ISM blueprint) 애니 및 머티리얼 제어

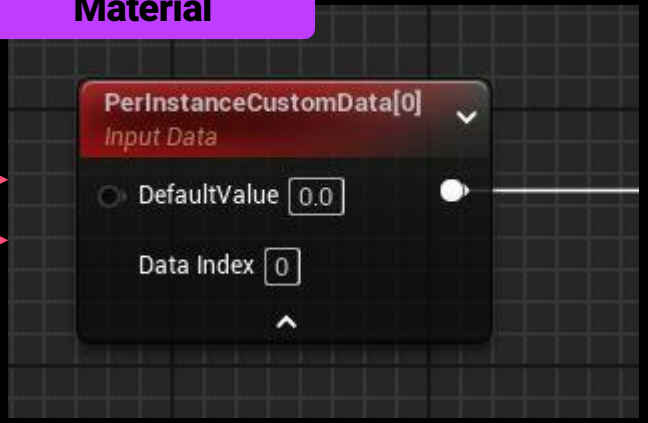
Update Animation



LED On

각 Instance 별로 Custom Primitive Data 값 지정

Material



Custom Primitive Data 값을 받아서 각 Instance의 Material 제어 (Anim Frame, LED emission)

감사합니다.