A character from Final Fantasy XIV, seen from behind, stands on a rocky shore looking out at a dramatic volcanic landscape. The sky is filled with dark, heavy clouds, with a bright orange and yellow sunset or sunrise glow on the horizon. Jagged, dark rock formations and a distant, partially submerged structure are visible in the background.

게임 엔진

# LEC 15 블루프린트 통신



한국공학대학교  
TECH UNIVERSITY OF KOREA

이대현 교수

# Blueprint Communication

---

- 여러 액터 들 사이에 작업을 시키거나, 정보를 전달하고 공유하는 것.
  - 다른 액터로 하여금 어떤 일을 하게 함 - 예) 방안의 불을 켜.
  - 다른 액터에게 정보를 전달함 - 예) 데미지 전달
  - 다른 액터의 상태, 속성, 변수값, 결과 등을 파악함.
  - 다른 액터들(Listener)에게 공통적인 내용을 전달함(Broadcasting).

# 블루프린트 통신(Blueprint Communication)

- 통신을 위해서는 보내는 Blueprint 한 개와, 최소한 한 개 이상의 받는 Blueprint들이 존재
- 모든 통신은 1-Way
  - 양방향으로 하려면, 각각 따로 설정이 되어야 함.
- 통신에는 반드시 Reference(참조 이름 - 특정 객체를 지정할 수 있는 이름)가 필요함.
  - Sender, Receiver 둘 중 하나는 상대를 알아야 함 - 예) 이름, 주소



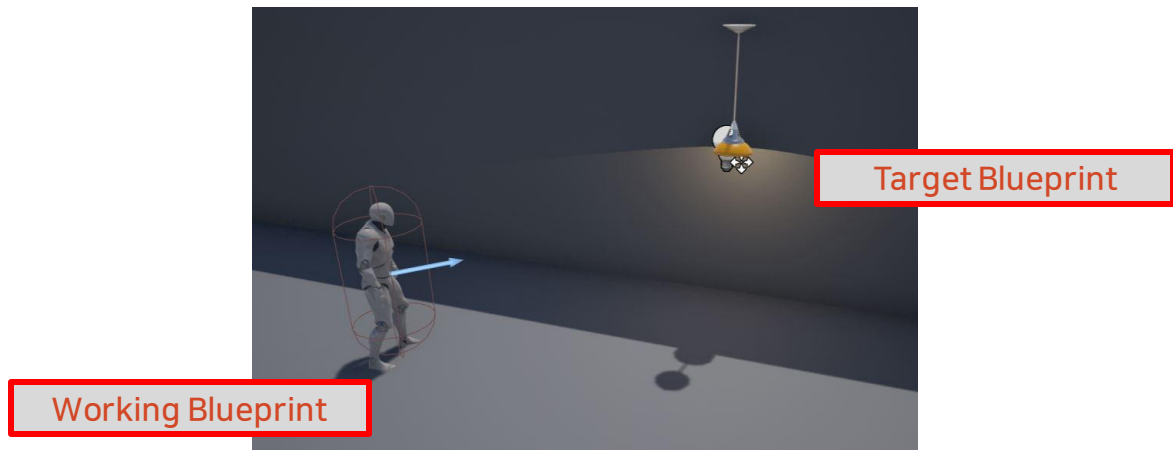
# 블루프린트 통신 유형

---

- 상대 액터의 참조를 획득하는 방법과 메시지의 전달 방법에 따라 분류
- 직접 통신(Direct Communication)
- 이벤트 디스패처(Event Dispatcher)
- 블루프린트 인터페이스(Blueprint Interface)

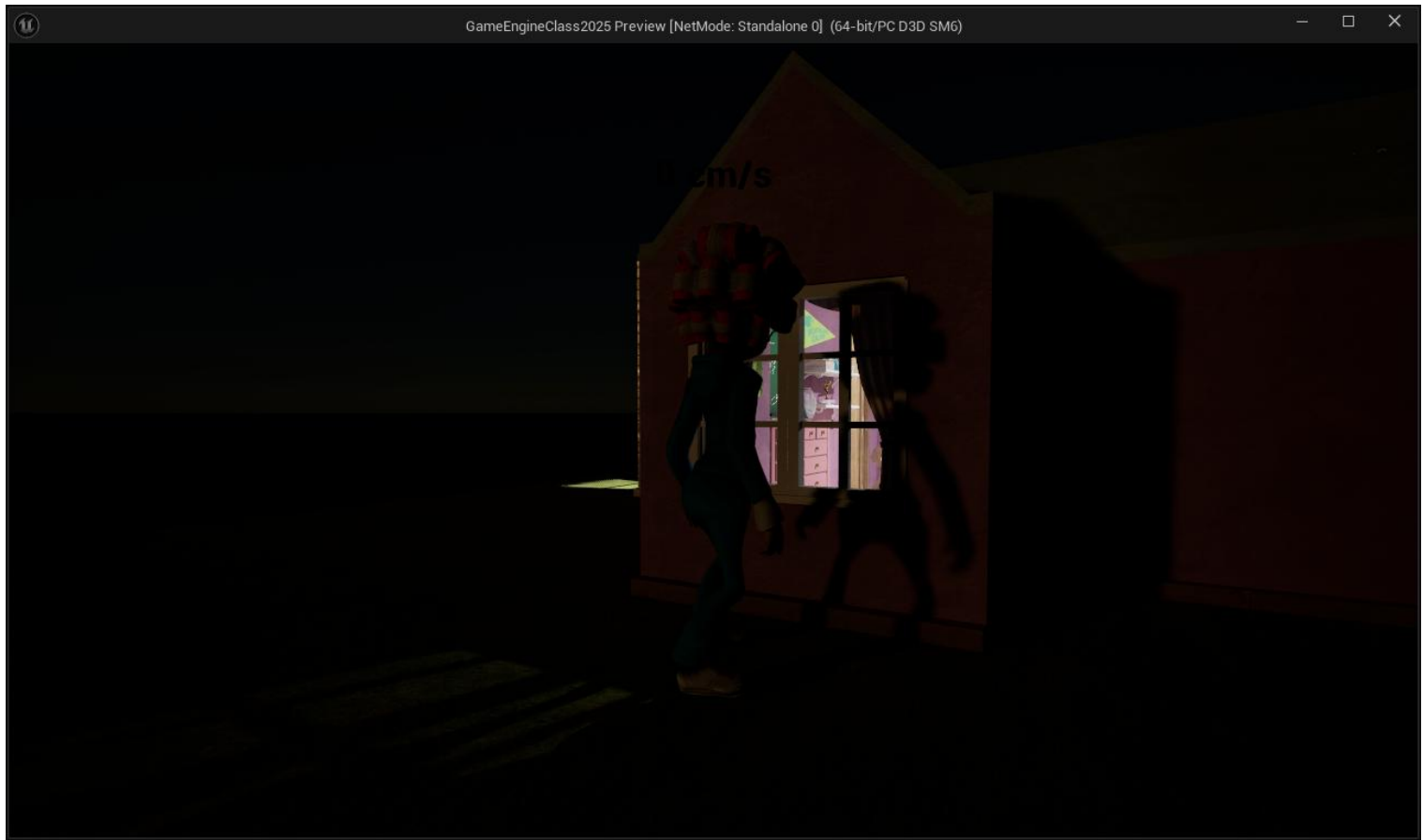
# 직접 통신(Direct Communication)

- 메시지를 주고받으려 하는 액터가 두 개 있을 때
  - Working Blueprint → Target Blueprint
  - One-to-One 통신
- Target Blueprint 를 쉽게 찾을 수 있을 때 유리
  - 액터가 맵 상에 이미 존재할 때, 또는 함수를 통해 쉽게 reference를 알 수 있을 때.

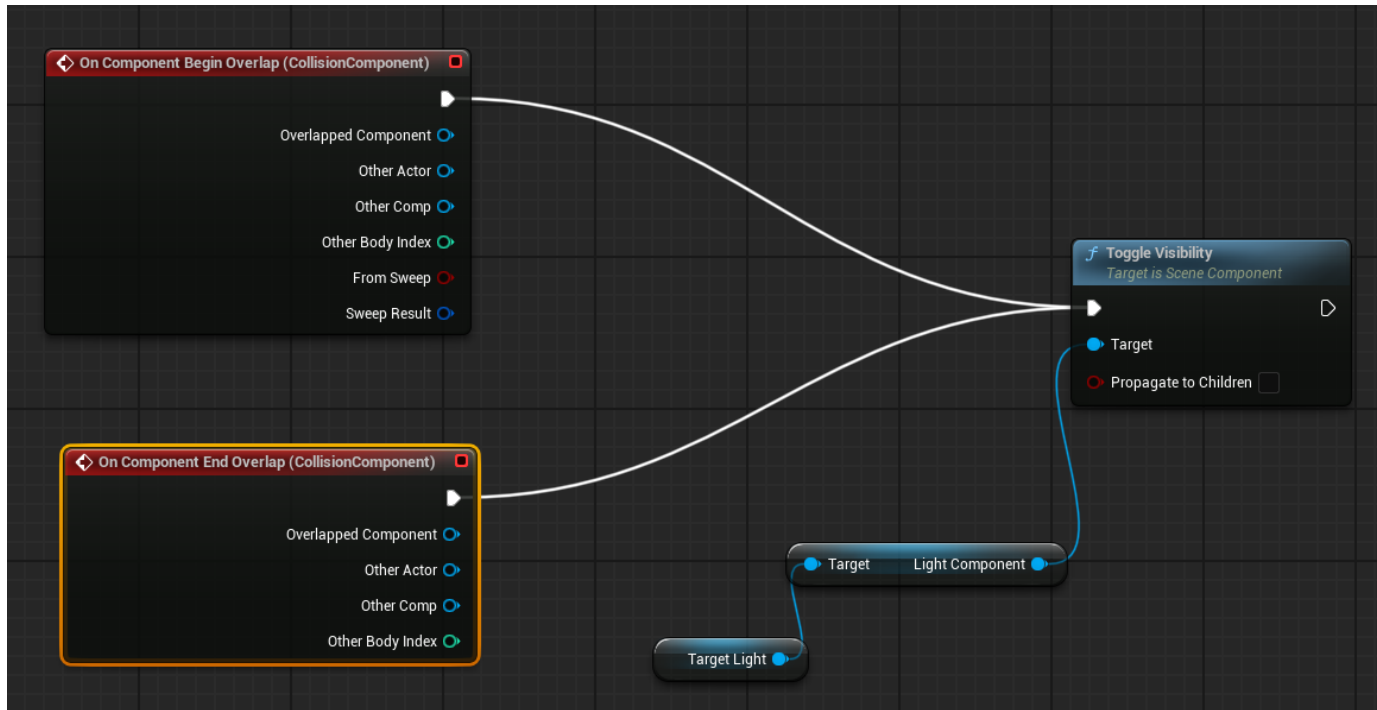




## 실습 방안의 불켜기



# Senor TriggerBox



라이트 액터를 Instance Editable 로 만들고, 레벨에서 액터를 Pick 연결



# Event Dispatcher

---

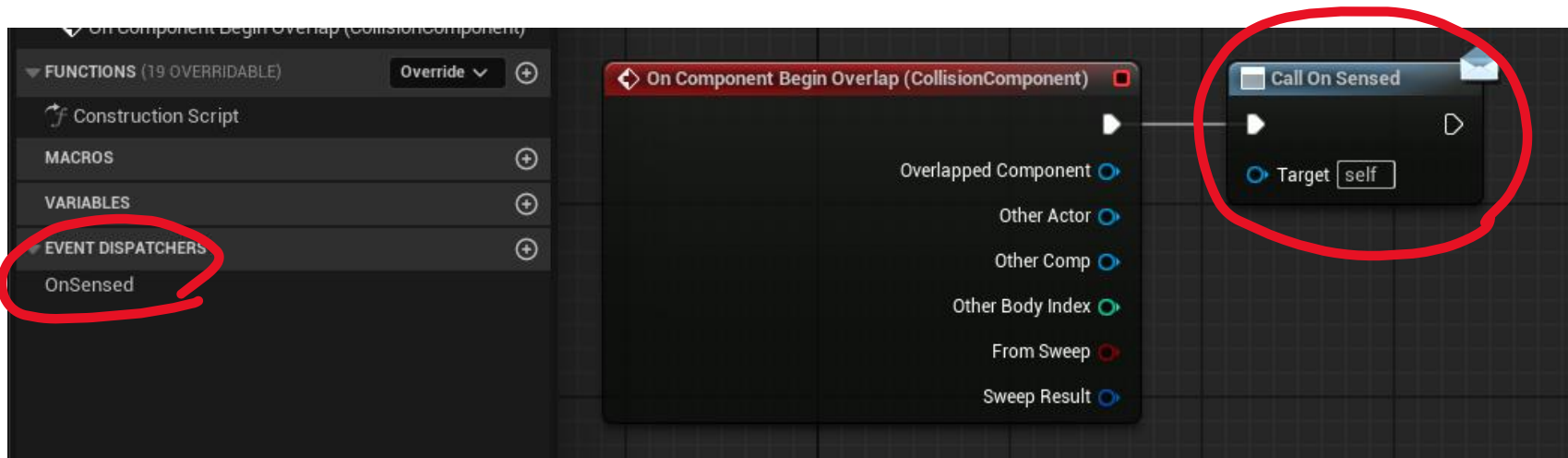
- 이벤트의 발생 여부를 대기 중인 다른 블루 프린트에게 알리기에 적합.
- 이벤트 발송에 따른 처리 여부를 실시간으로 변경할 수 있음.
- Event Dispatcher 가 필요한 경우.
  - 캐릭터 BP에서 Level Blueprint와 통신을 하고 싶어함.
    - 레벨 업함으로써, 레벨 안에 있는 기존의 잠긴 영역을 오픈.
    - 캐릭터가 레벨 전체의 객체들에게 동시다발적으로 어떤 액션을 지시함.
  - 스폰된 액터에서 상태 변화가 일어나면, 이벤트가 발동.
    - 보스를 스폰하고, 보스가 죽으면 월드에 보상이 스폰되는 이벤트가 발동.
    - 레벨에 아이템을 스폰하고, 픽업시 아이템과 캐릭터에게 알림.



실습

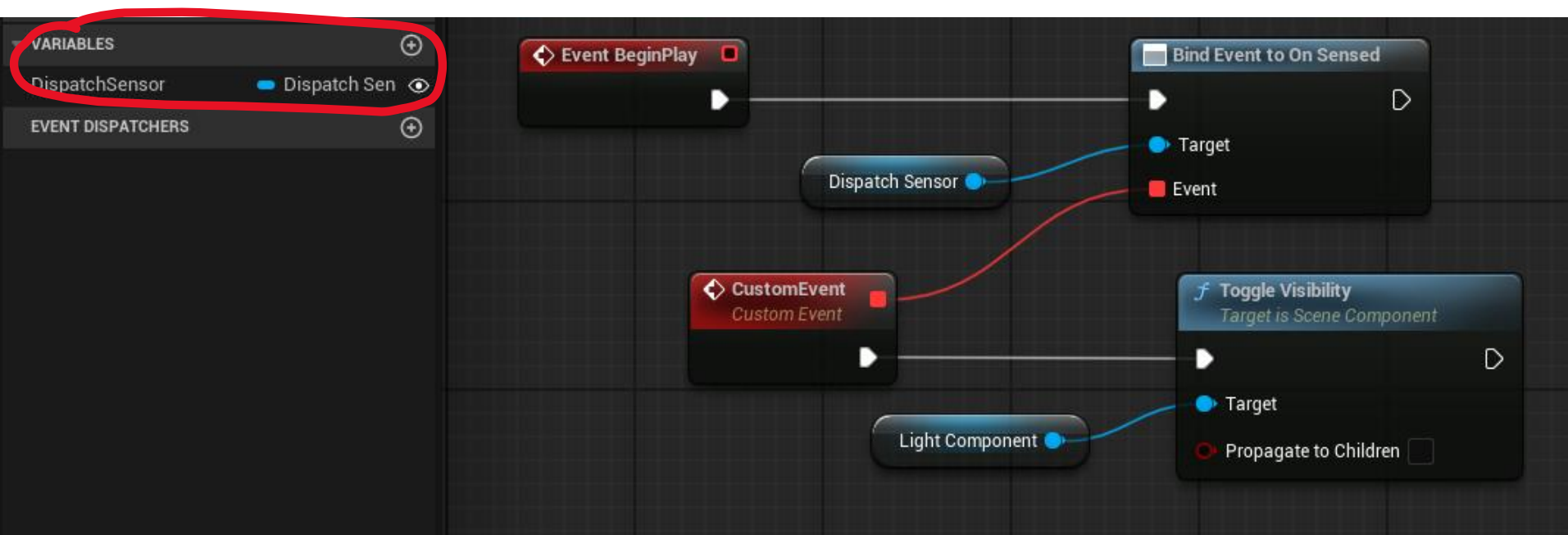
방안의 여러 개의 불 켜기

# 메시지 발송: Sensor Trigger Box



메시지를 발송하고자 하는 블루프린트에서  
메시지를 생성하고 호출

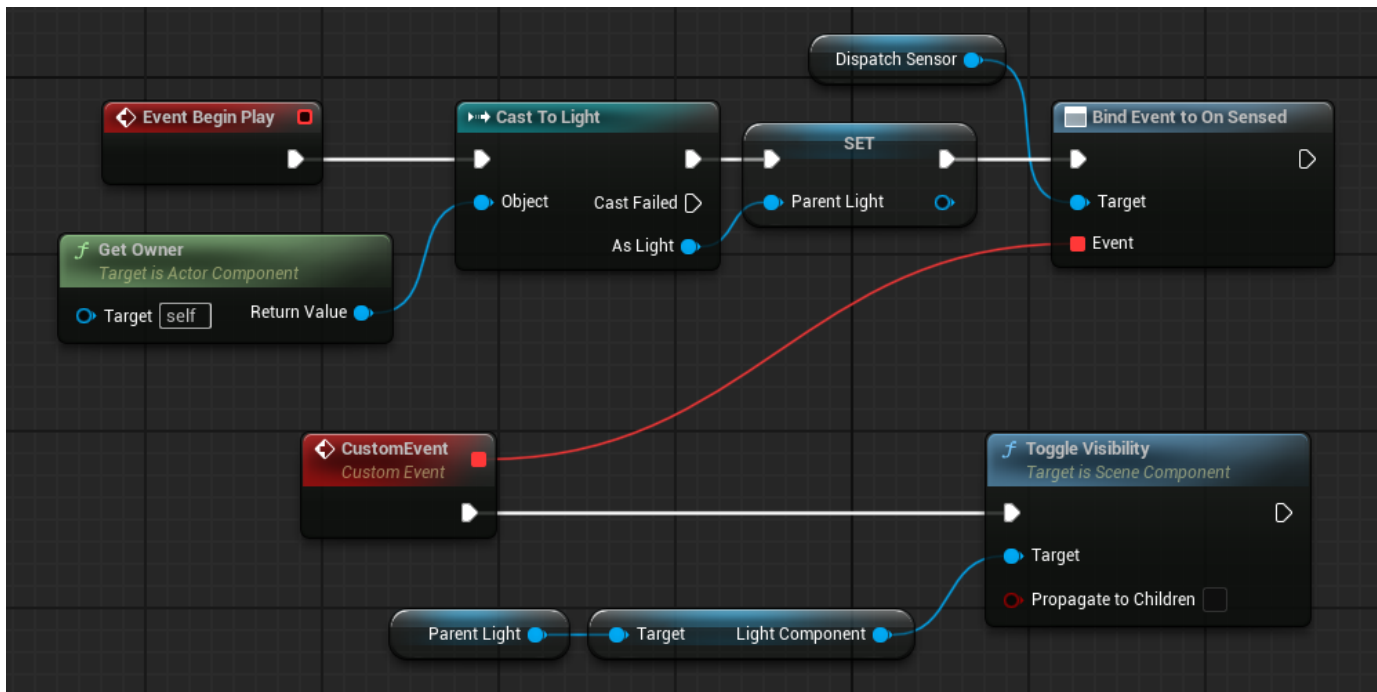
# 메시지 수신 : 라이트



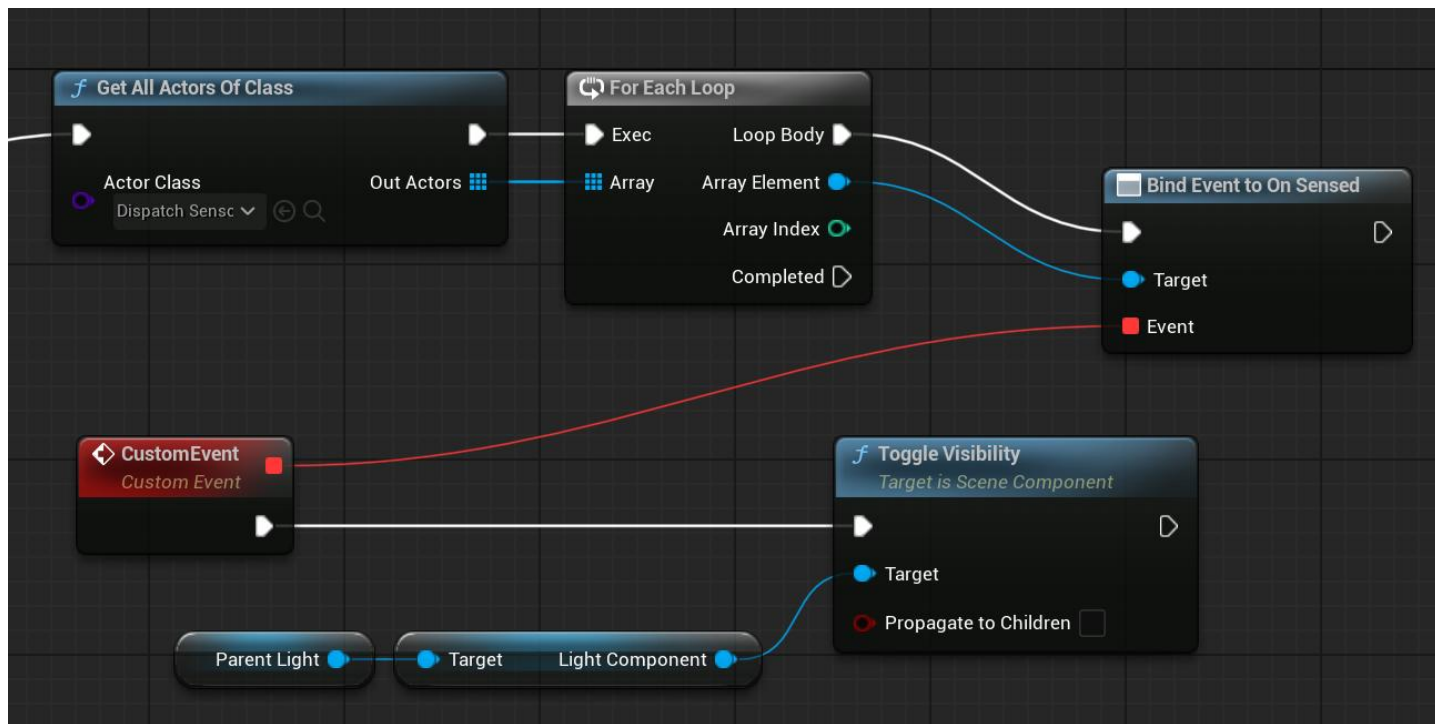
메시지 발송자의 레퍼런스를 확보한 후, 해당 메시지에 대해서 내부 이벤트를 만들어서 바인딩

# LightControllerComponent

- 센서의 OnSensed 에 대해서 라이트를 켜는 컴포넌트를 제작해서 기존 라이트 액터에 부착.

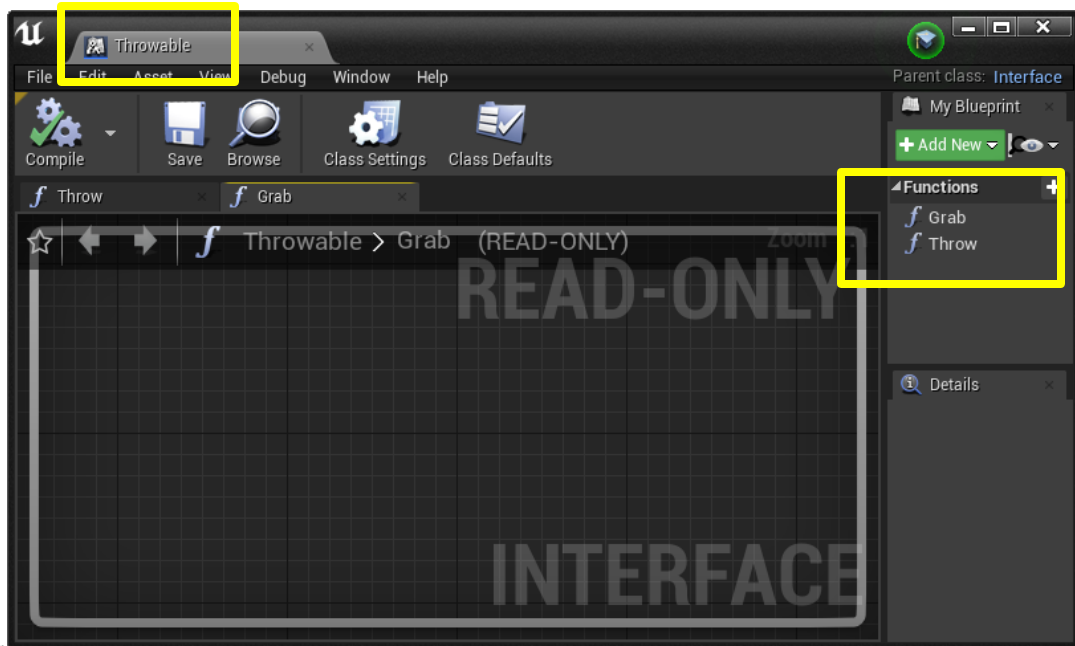


# 센서가 여러 개일 때는 ?



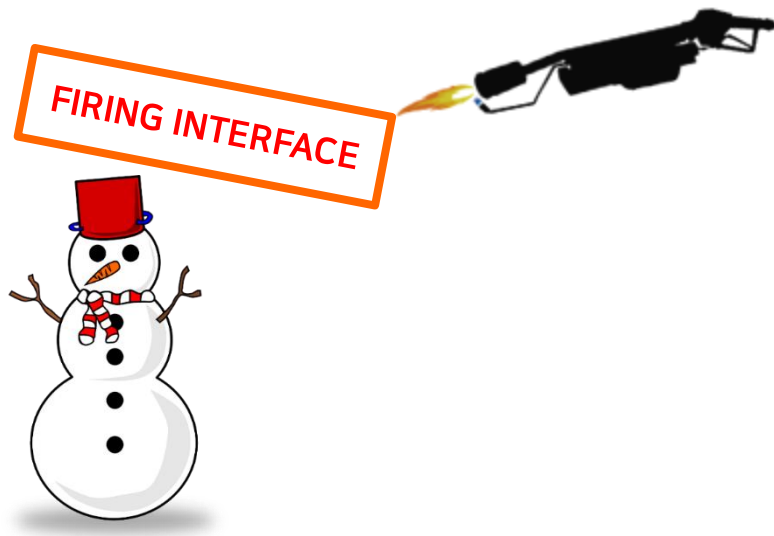
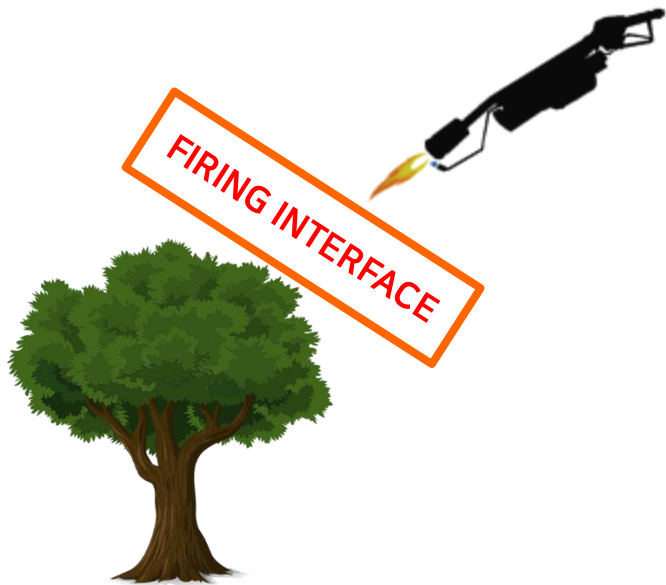
# 블루프린트 인터페이스

- 이름만 있고 구현은 없는 함수 하나 이상의 집합.
- 다른 블루프린트에 추가할 수 있음.
- 인터페이스가 추가된 블루프린트는 그 안에 함수를 갖게 됨.



# 블루프린트 인터페이스

- 객체 들에게 어떤 기능을 공통적으로 부여할 때 사용.
- 대상 객체와의 블루프린트 통신이 가능하게 해 줌.
- JAVA 언어의 Interface
- C++ 언어의 Abstract function





# 인터페이스의 역할



현재 공을 잡거나 던질 수 없음.

# 인터페이스의 역할



Throwable Interface를 Ball에  
추가하고, Throw 및 Grab 함수를  
구현해주면, 공을 잡고 던질 수  
있게 됨.

# 인터페이스의 역할



Throwable Interface를 의자에  
추가하고, Throw 및 Grab 함수를  
구현해주면, 의자를 잡고 던질 수  
있게 됨.

# 블루프린터 인터페이스

## ■ 일종의 계약

- “B, 너가 이 인터페이스를 구현하며(또는 함수로 구현), 내(A)가 호출하면 거기에 반응함.”
- B에게 인터페이스 구현이 없으면, 무시됨.

A



B





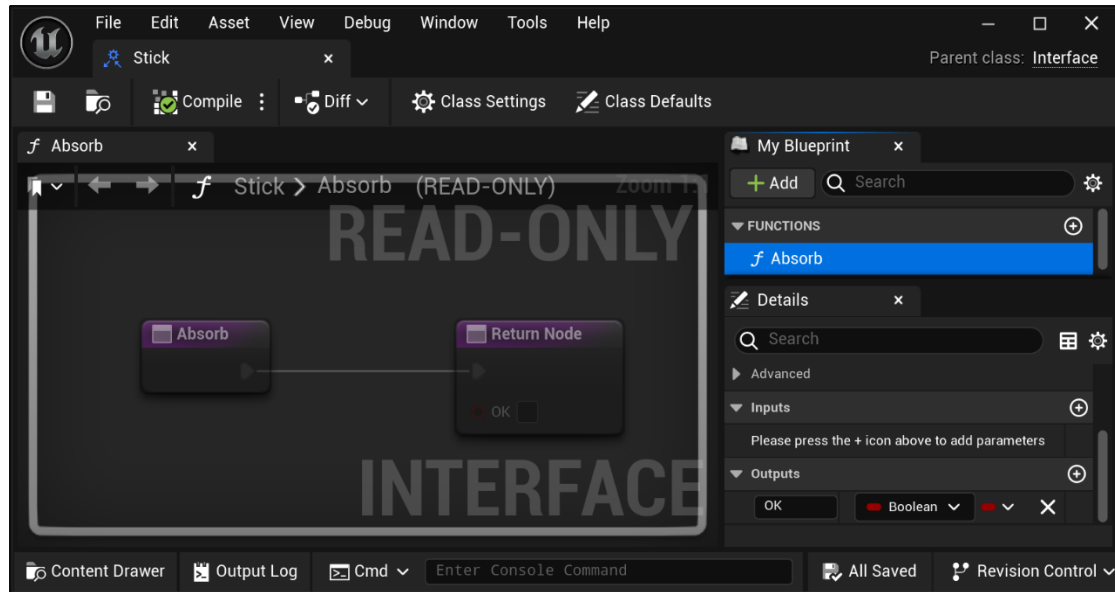
## 실습 물건 부착 구현

# 접착 인터페이스

물체를 붙인다는 의미로 인터페이스이므로  
이름은 "Stick" 라고 정합니다.

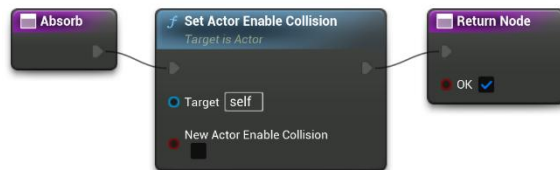


할머니가 인형 카트를 붙이려면 우선  
흡수해야 합니다. 따라서 이 요청을  
Absorb 라고 이름붙입니다.



# StickableMeshActor에서 Absorb 함수 구현

액터가 할머니에게 흡수되면 더 이상 충돌이  
발생하지 않도록 설정합니다.

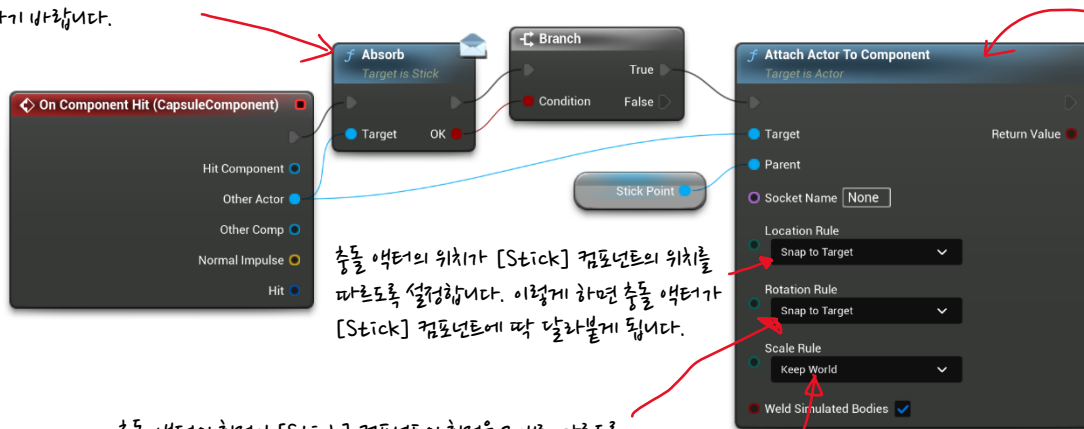


# Granny 에서 Absorb 함수 호출

이벤트 노드의 [Other Actor] 필드를 드래그한 후 [Absorb] 함수를 찾아 선택하면 됩니다. 만약 함수가 검색되지 않으면 [Stick] 인터페이스를 컴파일하고 저장했는지 확인하기 바랍니다.

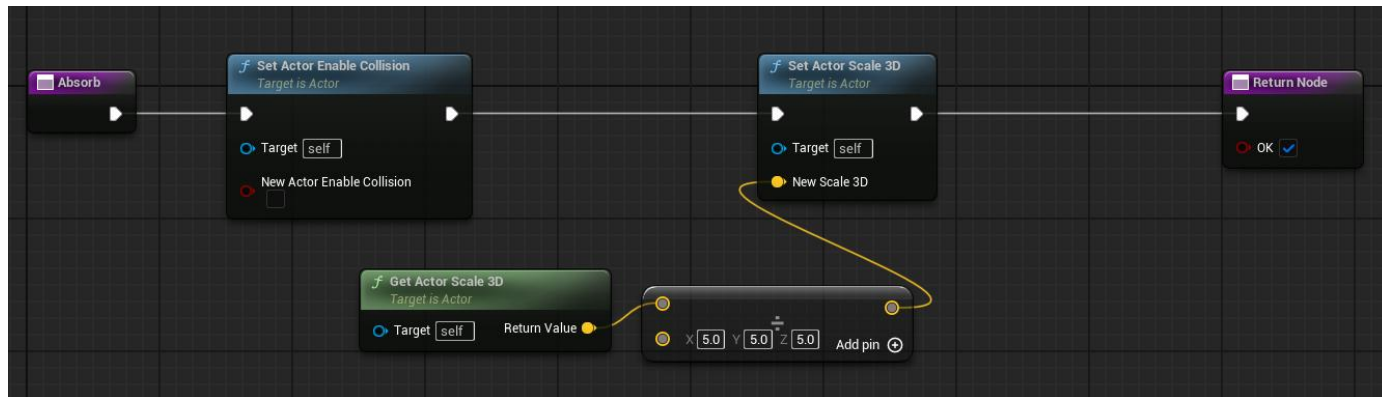
[Absorb] 함수 호출 결과 [OK]가 참이면, 즉 정상적으로 [Absorb] 함수의 수행이 이루어졌다면 충돌 액터를 [StickPoint] 컴포넌트에 갖다붙입니다.

[Attach Actor To component] 함수는 [Parent] 컴포넌트에 [Target] 액터를 부착합니다.

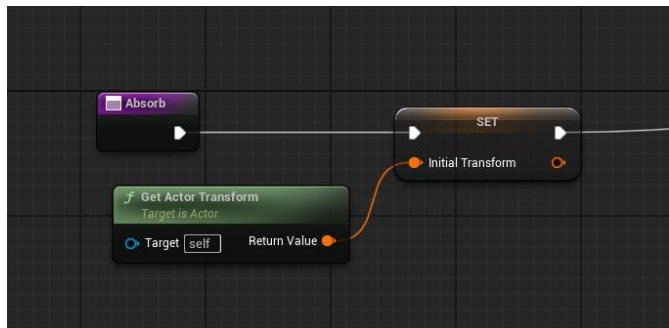




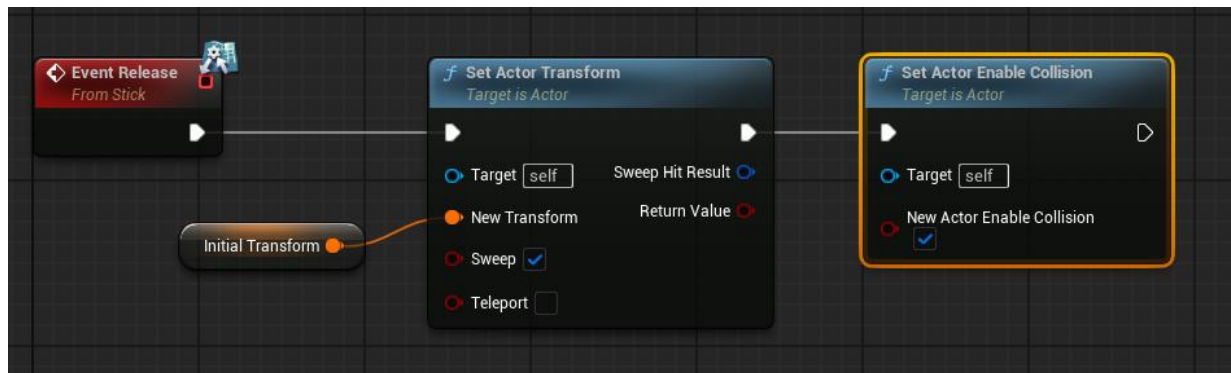
# 흡수 시 사이즈 축소



# 방출 Release 함수

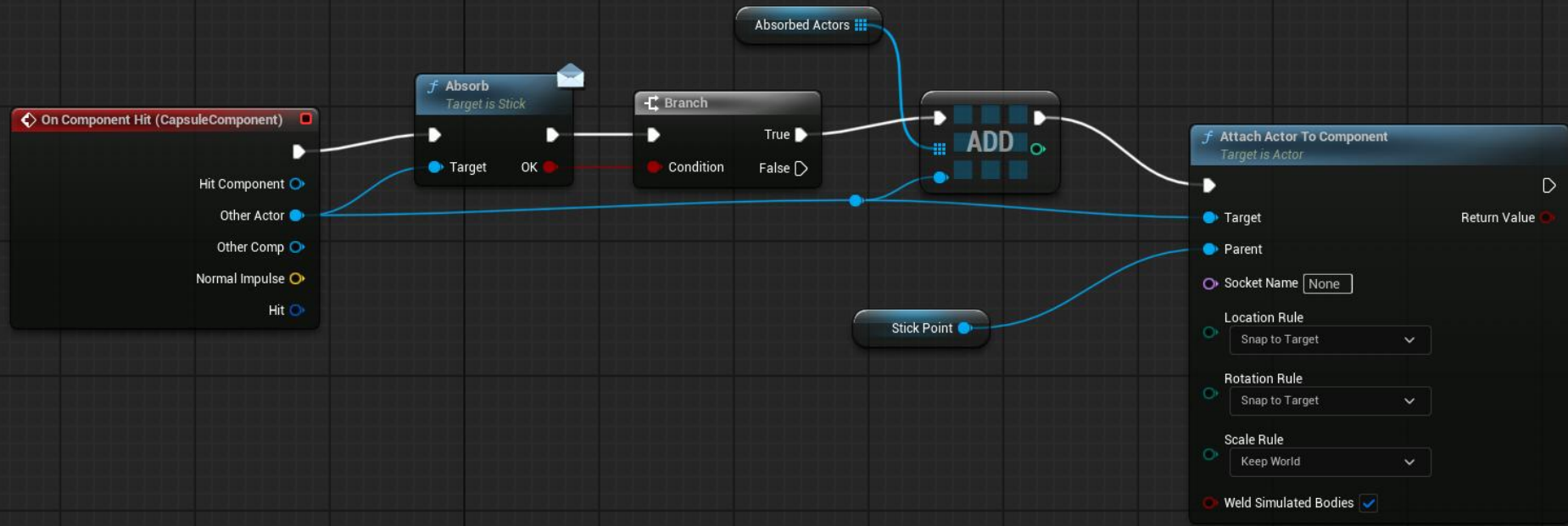


흡수 시, 기존 transform 을 저장

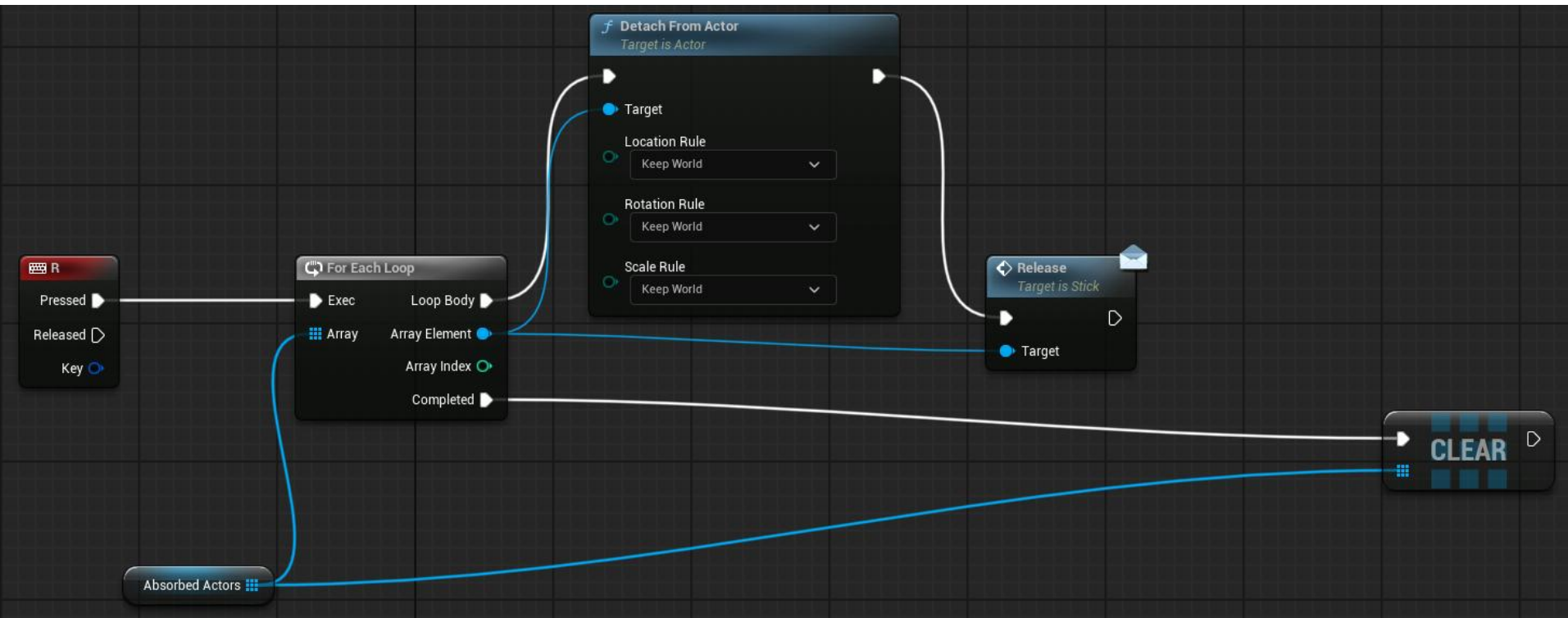


방출 시, 기존 Transform 복구 및 충돌 복구

# 흡수 한 모든 액터들을 저장

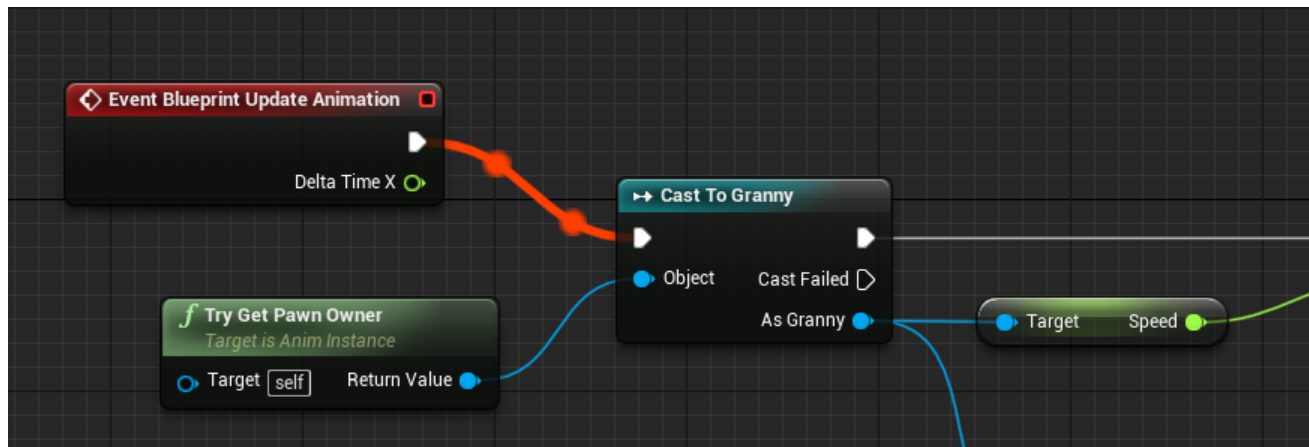
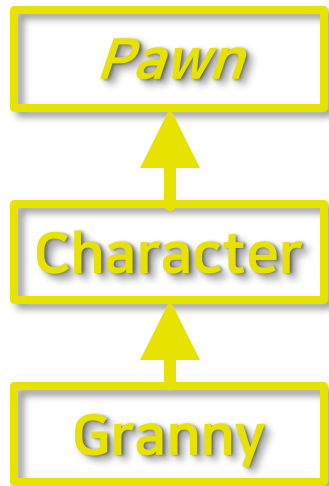


# R 키에 따른 방출



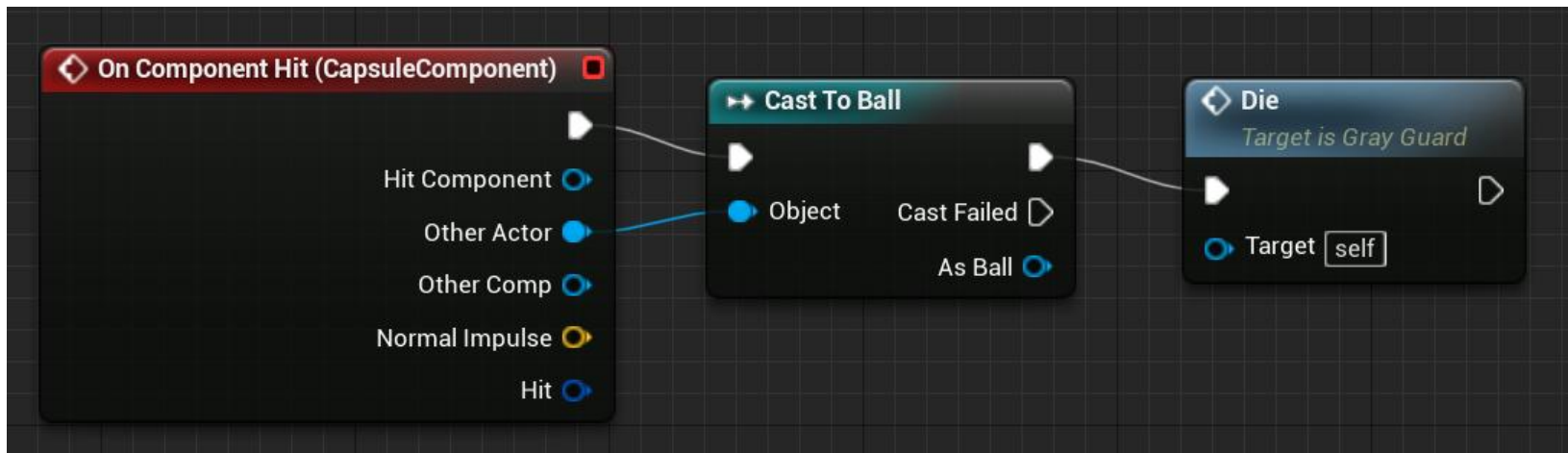
# 블루프린트 형변환(casting)

- 객체의 타입(클래스)을 특정 타입(클래스)으로 변경하는 것.
- 상속을 통해서 만들어진 자식 블루프린트 안에서 만들어진 내용을 액세스할 때 활용.
- 직접 통신을 할 때, 경우에 따라서, casting 이 필요.



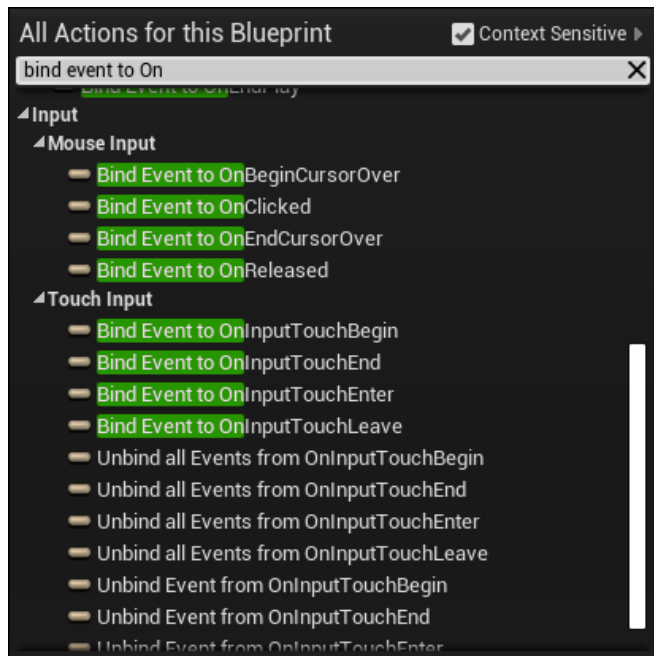
# 블루프린트 형변환(casting)

- 형변화를 하려는 액터가 형변환 대상 액터와 같은 검사할 때도 사용할 수 있음.
- 느리기 때문에, 틱에서 호출하는 것은 피해야 함. 미리 다른 변수에 저장해 놓는 것이 유리함.



# 내부 이벤트 발송

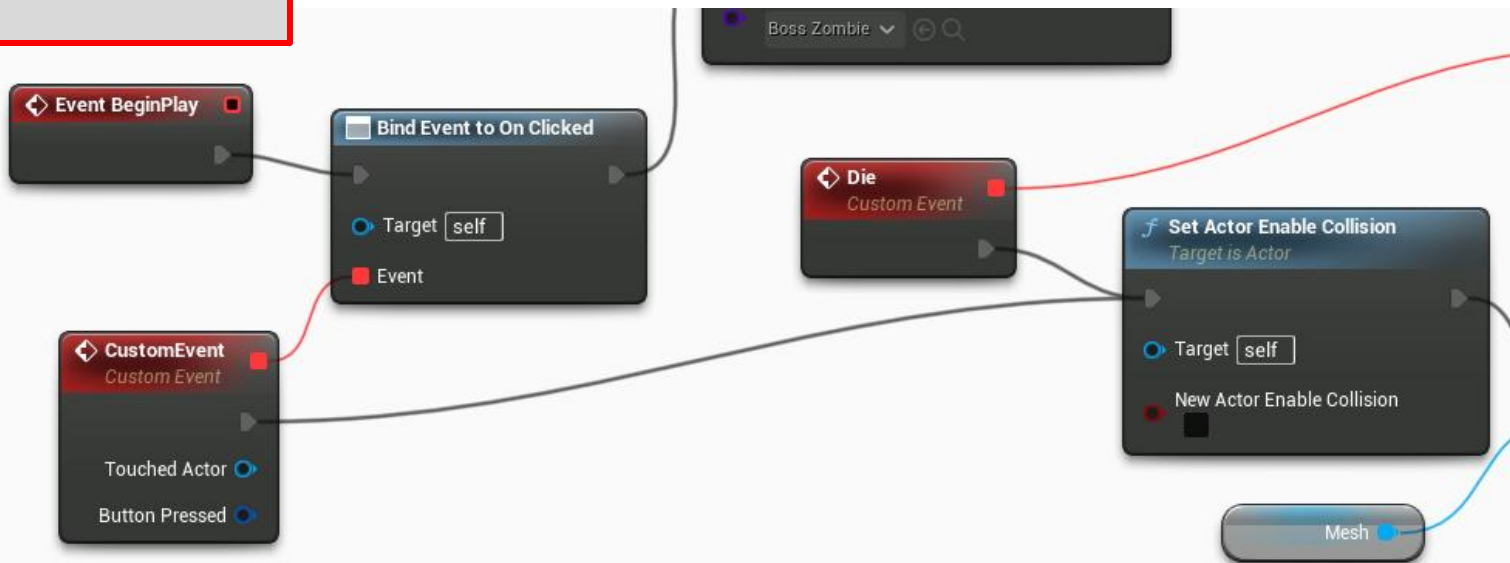
- 언리얼 엔진은 액터들이 다양한 이벤트 발송들을 내부적으로 이미 갖고 있음.
- 새로운 이벤트 발송을 만들기 전에 미리 기존 이벤트 발송들을 확인해야 함.
  - 예) 어떤 액터를 클릭했을 때, 이에 따른 처리를 하려면?



PlayerController



zombie





# 요약

---

## ■ 블루프린트 통신

- 독립적인 객체들이 서로 상호작용하기 위한 방법

## ■ 통신 유형

- 직접 통신(Direct Communication) A → B
  - 상대방 객체에 대한 직접적인 레퍼런스를 확보한 후, 통신
- 블루프린트 인터페이스(Blueprint Interface)
  - 인터페이스가 필요한 객체가 있으면, 그 객체에 인터페이스를 추가하고, 그 인터페이스 함수를 구현.
- 이벤트 디스패처(Event Dispatcher)
  - 여러 객체들에게 이벤트를 방송(Broadcasting)함.
  - 1 대 다 통신