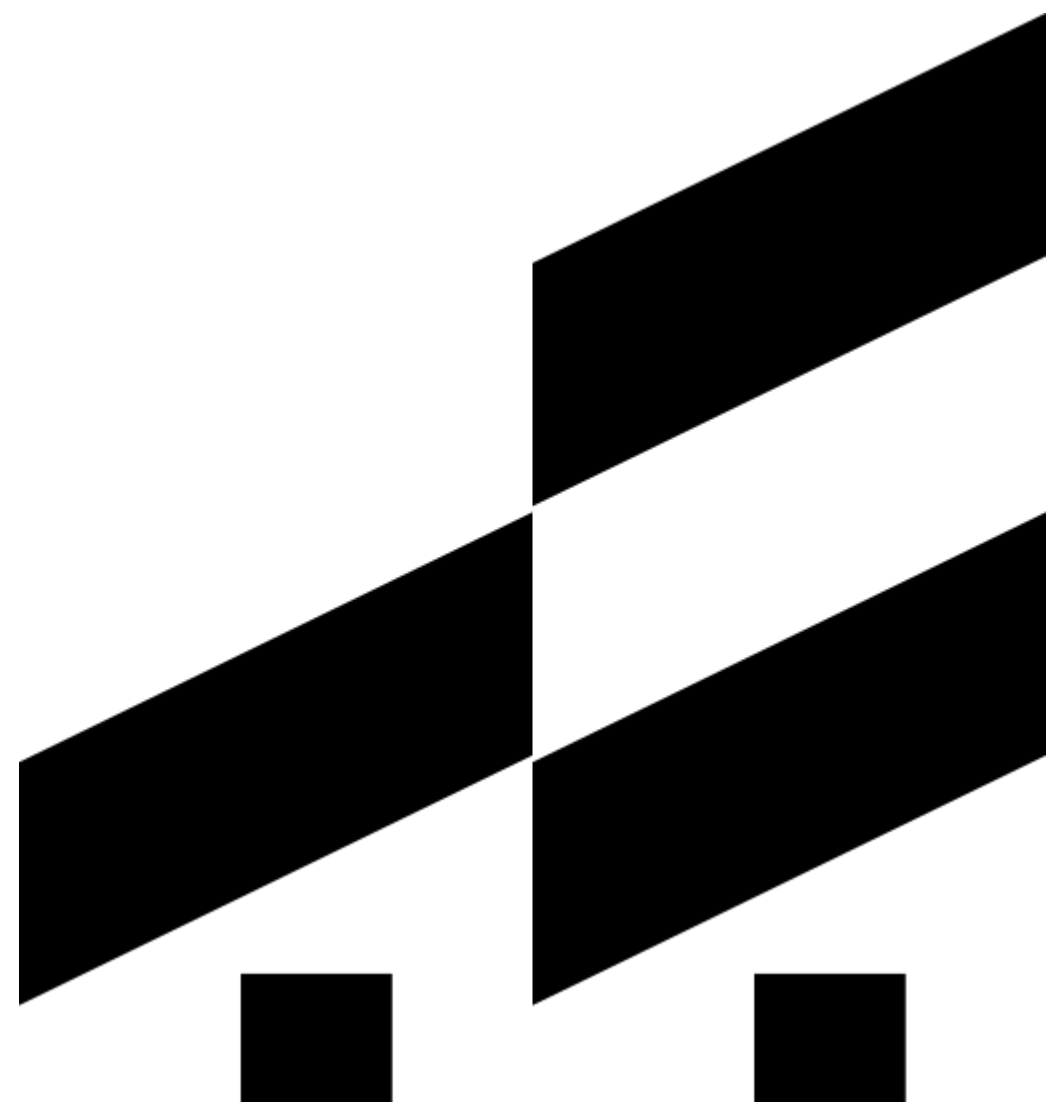


910005

포트폴리오



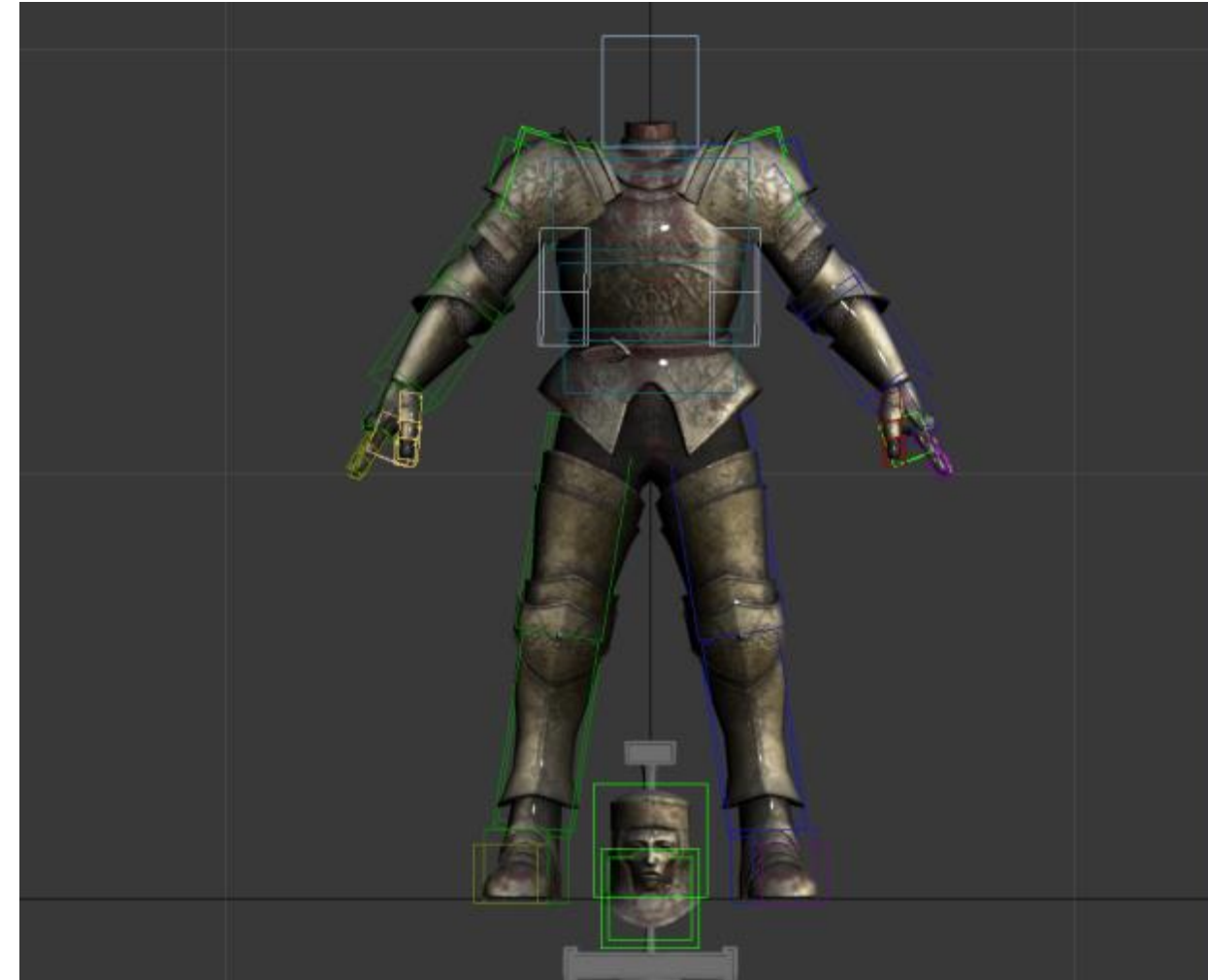
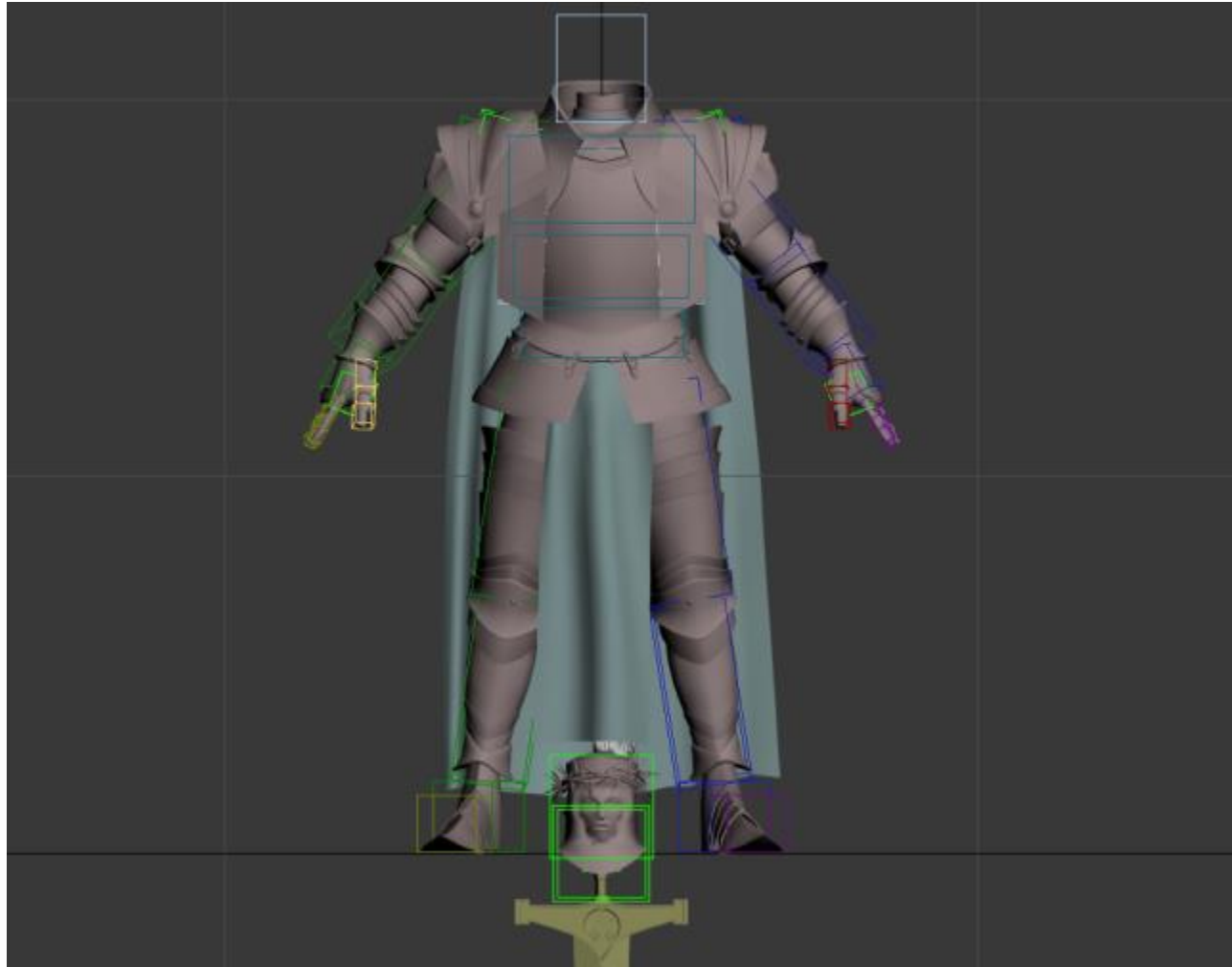
01 Rigging

02 MaxScript

01. Rigging



01. Rigging



Elite Knight & Normal Knight

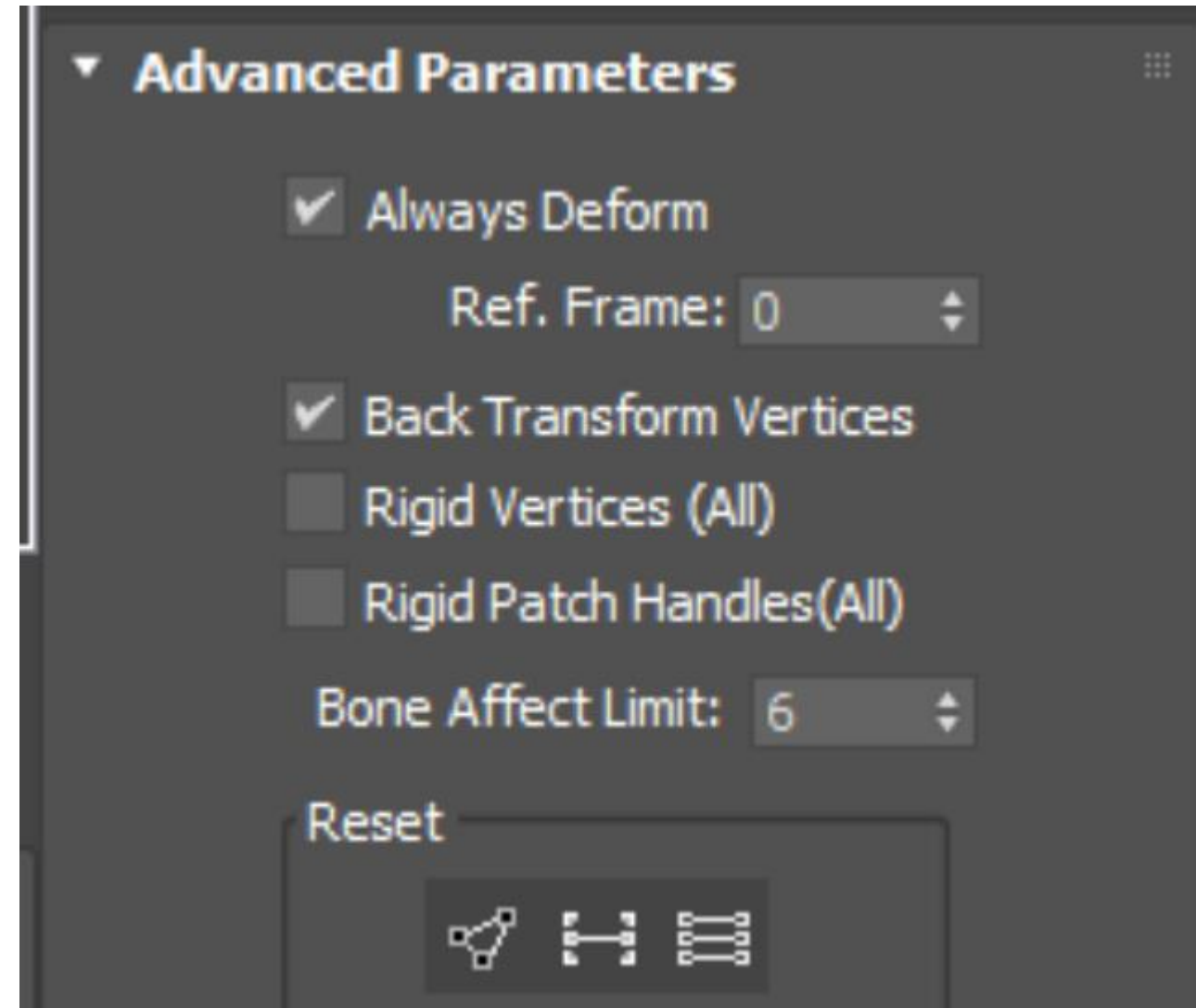
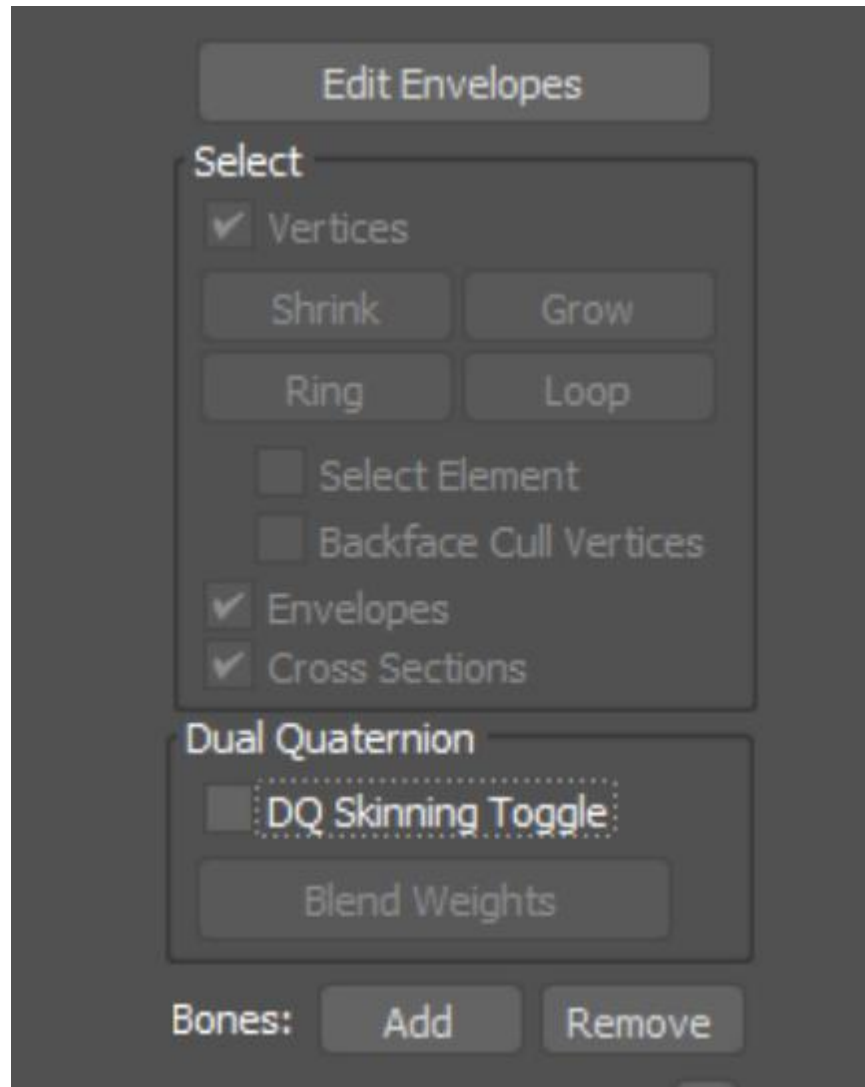
인게임에서 등장하는 적 몬스터로 플레이어의 적 참수 기능 구현을 위해 헤드를 분리해서 세팅 했었던 캐릭터입니다.

엘리트 기사(좌) 와 일반 기사(우) 가 같은 스켈레탈과 애니메이션을 공유합니다.

해당 캐릭터의 경우 기획 상 동일한 지역에서 여러마리가 소환되거나 하는 부분이 있기 때문에

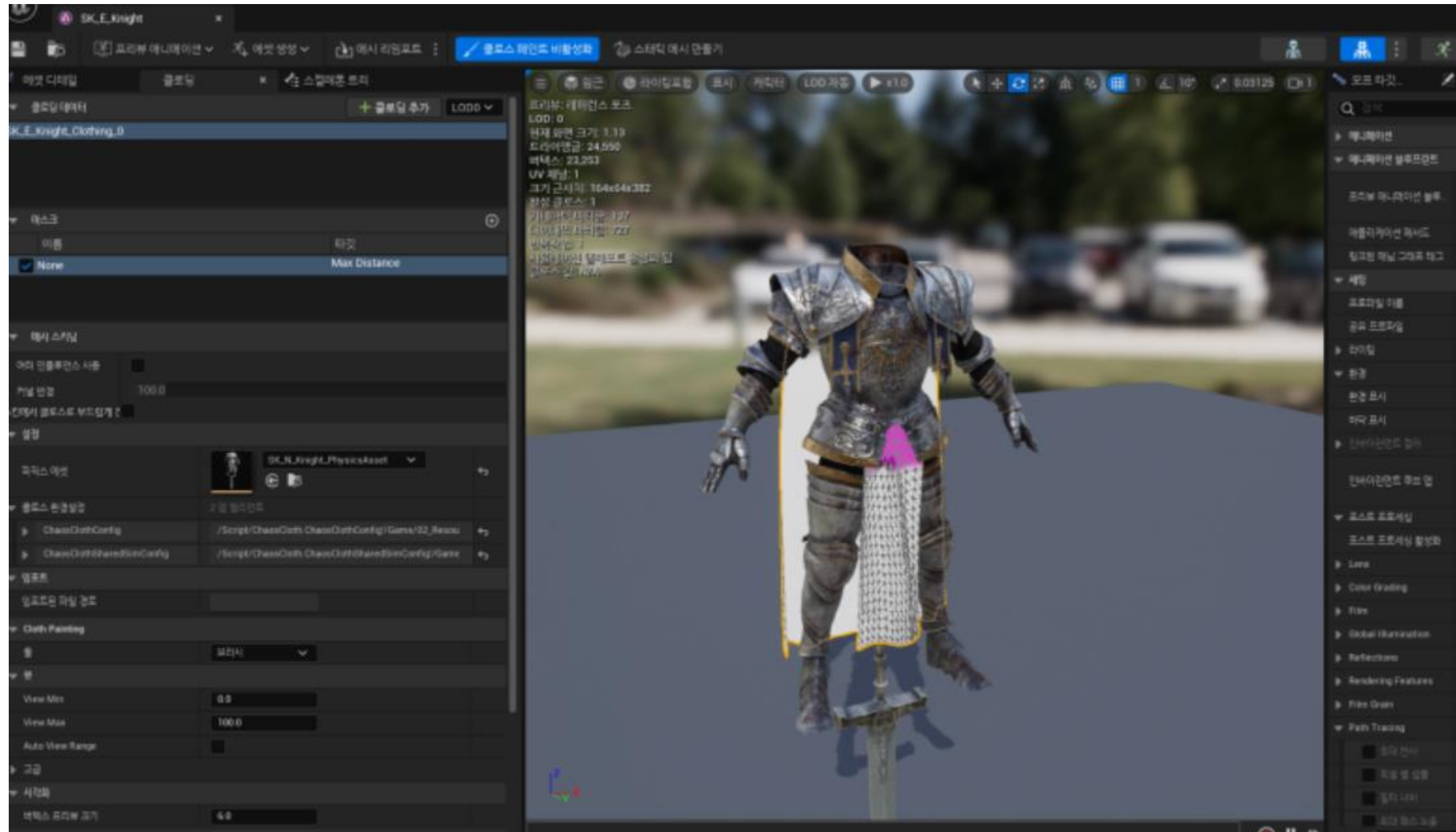
bone과 폴리곤을 비교적 적게 사용하여 제작되었습니다.

01. Rigging



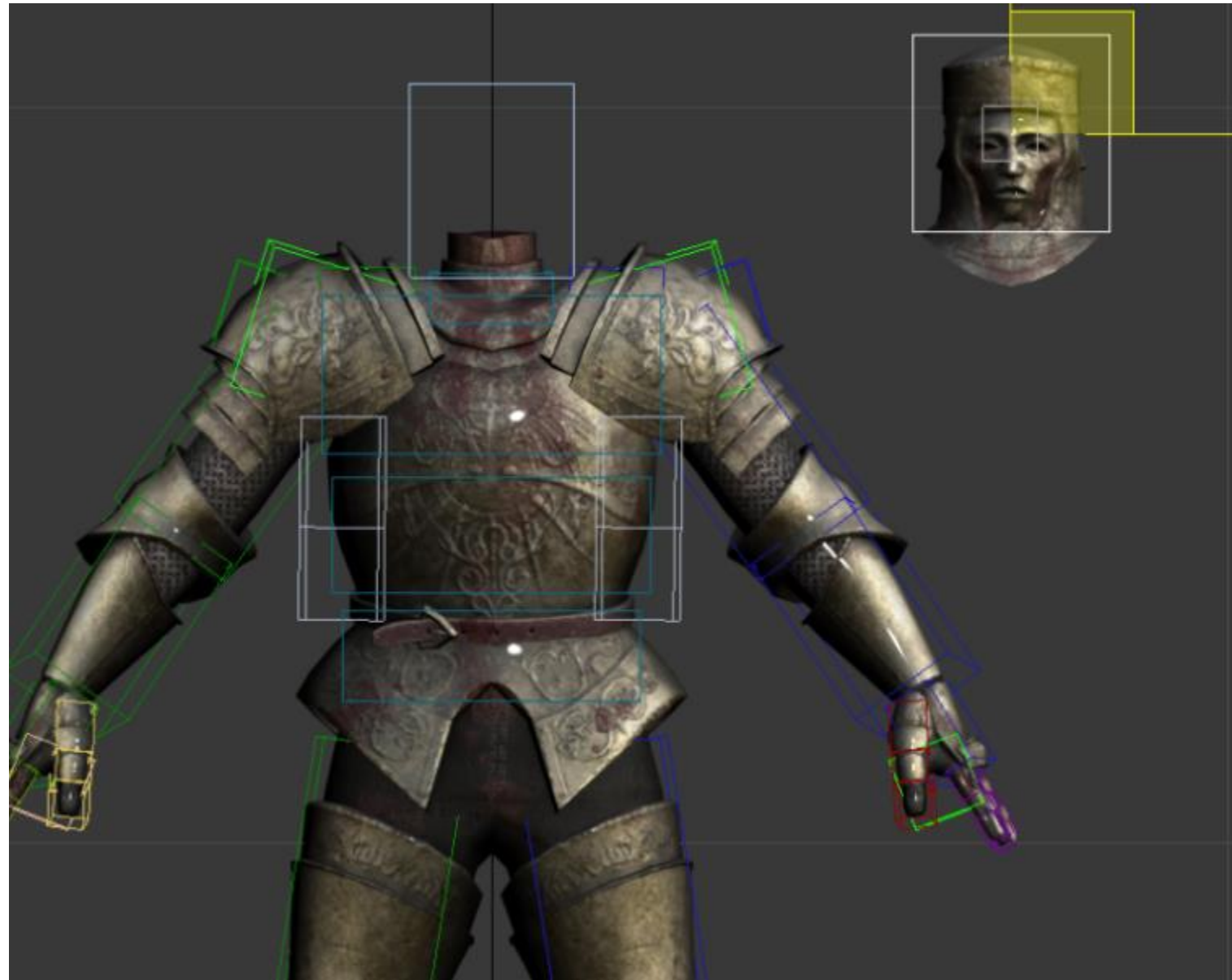
언리얼엔진에 임포트 되는 게임용 리소스임으로 dq를 비활성화 하고 bone affect limit을 6으로 설정하였습니다.

01. Rigging



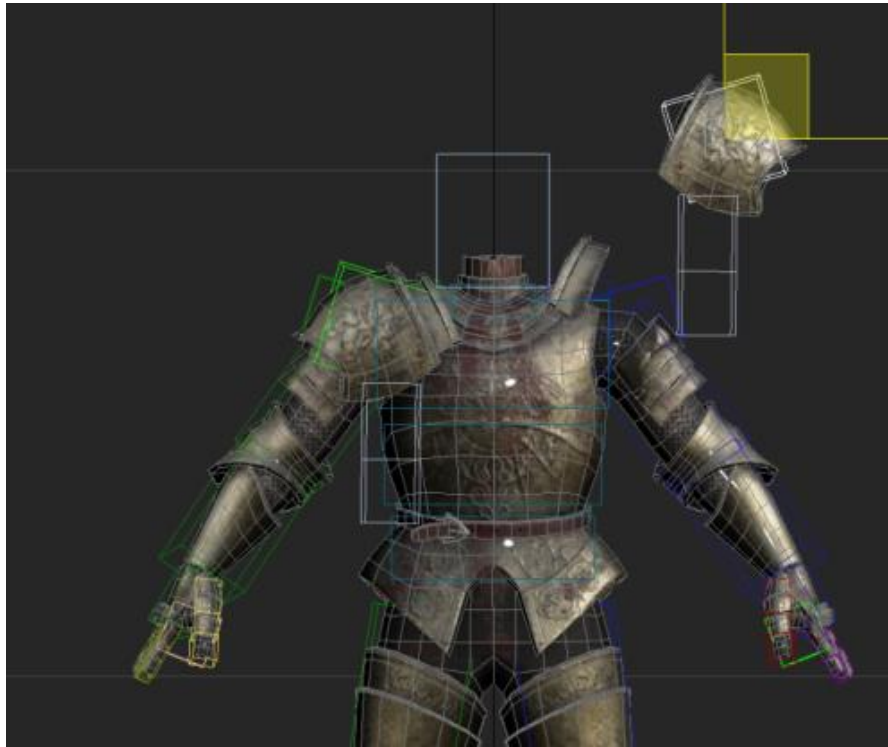
망토와 치마에 해당하는 부분에 bone을 세팅하지 않고 언리얼 엔진상에서 cloth툴을 사용하였습니다.

01. Rigging



헤드메쉬와 헤드메쉬용 본을 따로 만들고 각각 엔진에 독립된 개체로 임포트 한 뒤
몬스터 bp 내부에서 align 시키고 참수에 관한 명령어를 input 값으로 받으면
머리가 떨어지는 애니메이션을 재생하면서 분리 시키기 위한 구조를 설계 하였습니다.

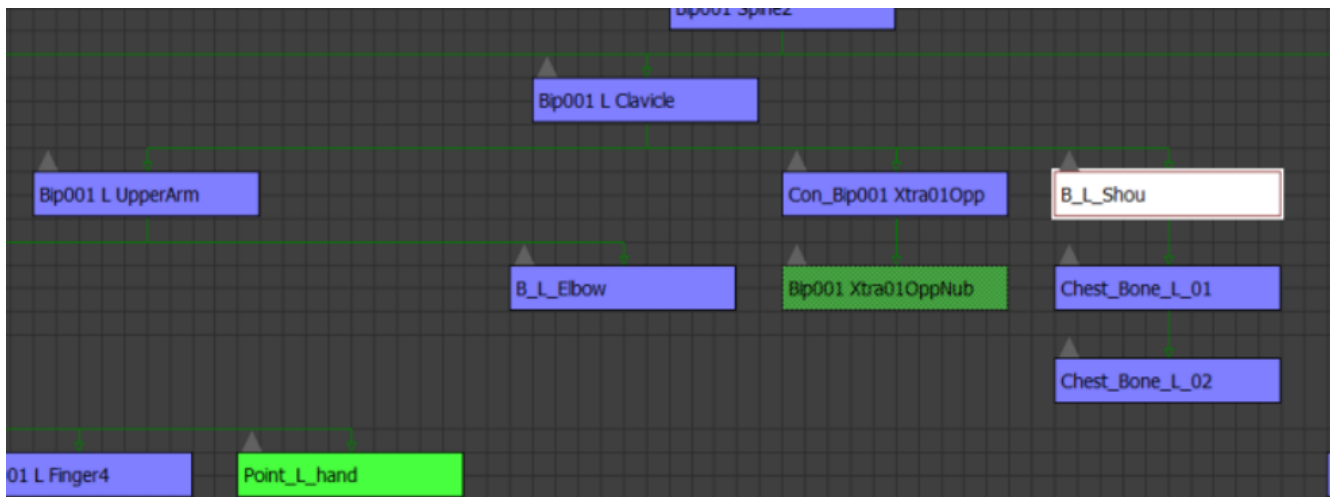
01. Rigging



어깨 갑옷리깅의 경우 효율적인 키애니메이션을 위해 포즈를 저장 하기 수월한 bipad의 xtra bone을 기본 베이스로 하고

해당 xtra bone의 움직임을 참조하여 따라하는 포인트 헬퍼를 설치하였습니다.

애니 작업시에는 좌우 각 2개씩 도합 4개의 본을 사용하는 구조입니다



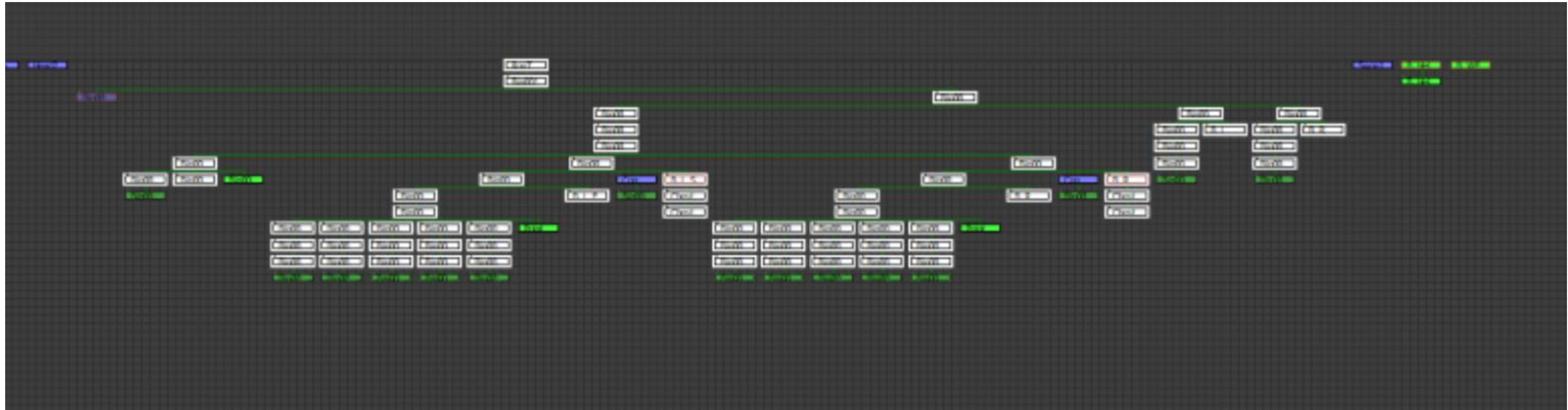
그러나 익스포트시엔 xtra를 제외하고 포인트 헬퍼만 BONE으로 익스포트하기에

최종적으로 엔진에서 사용하는 본은 2개 였습니다.

다른 캐릭터들에 비해 비교적 단순한 세팅이었기에 작업에 많은 시간을 소모하지는 않았지만

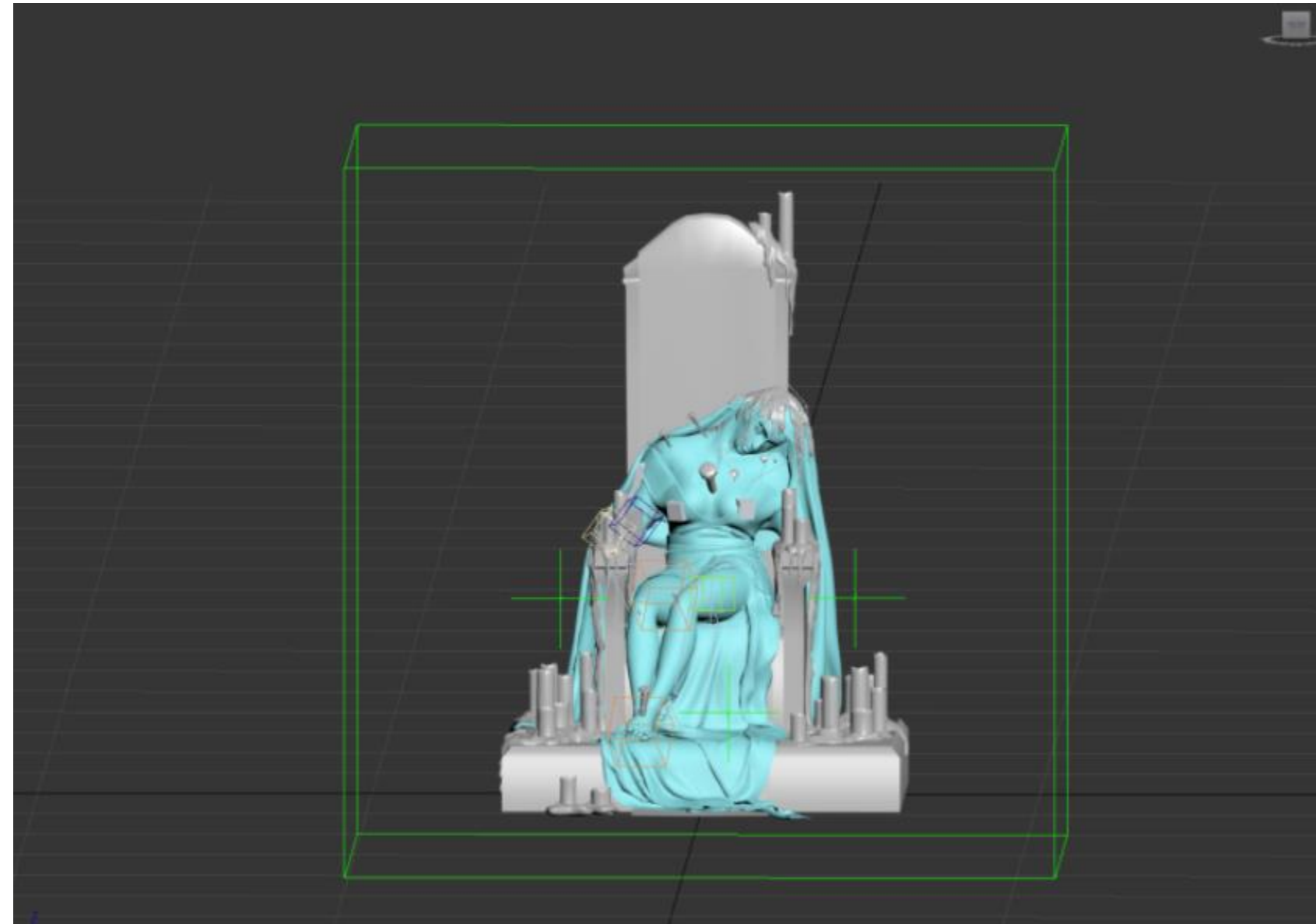
엔진에서 모델링만 다르고 스켈레탈을 공유하는 몬스터를 만든다는 점이 흥미로웠고 신선한 작업 이었던 것 같습니다.

01. Rigging



최종 임포트시 사용하는 본은 총 65개 입니다.

01. Rigging



Asabel

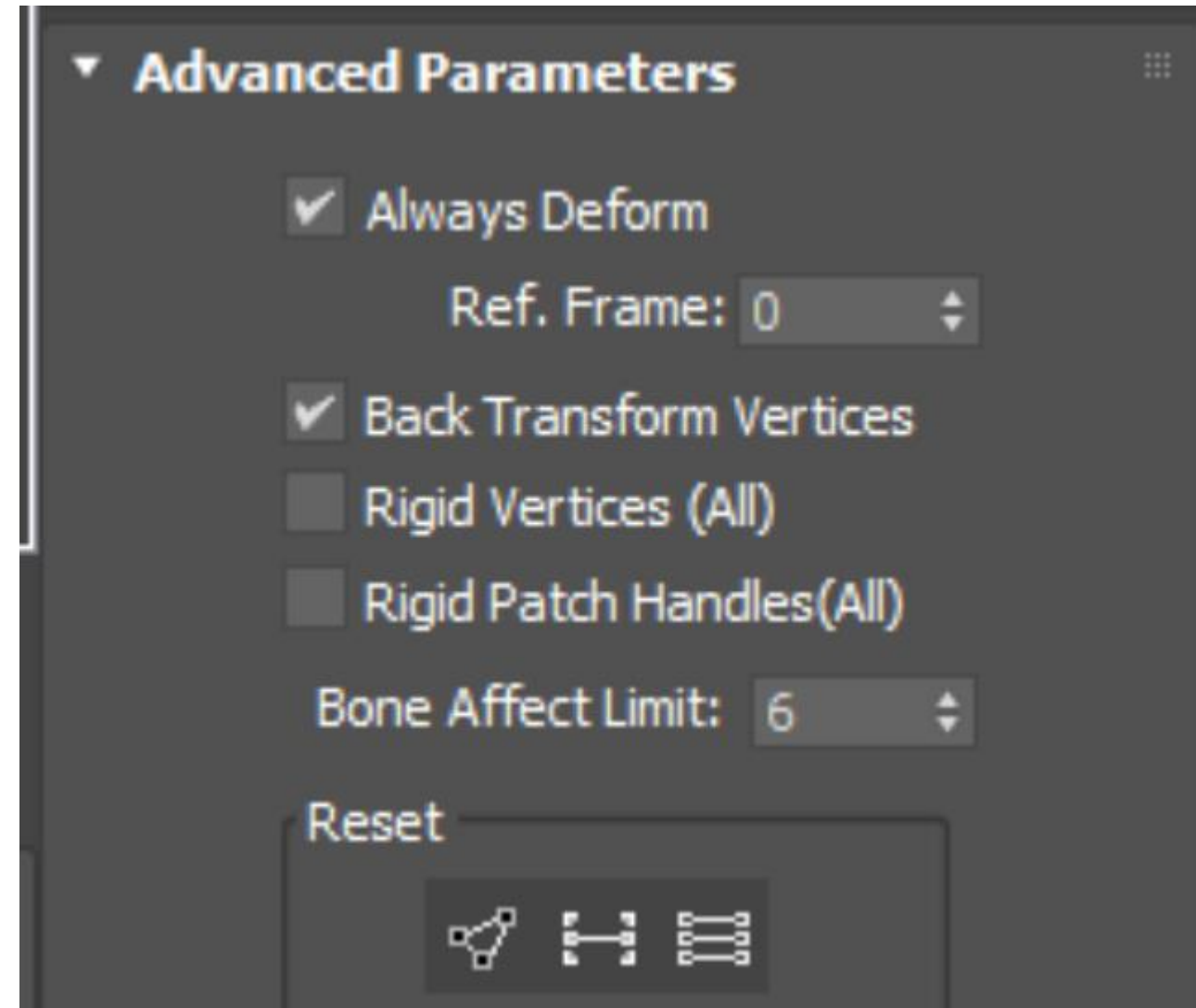
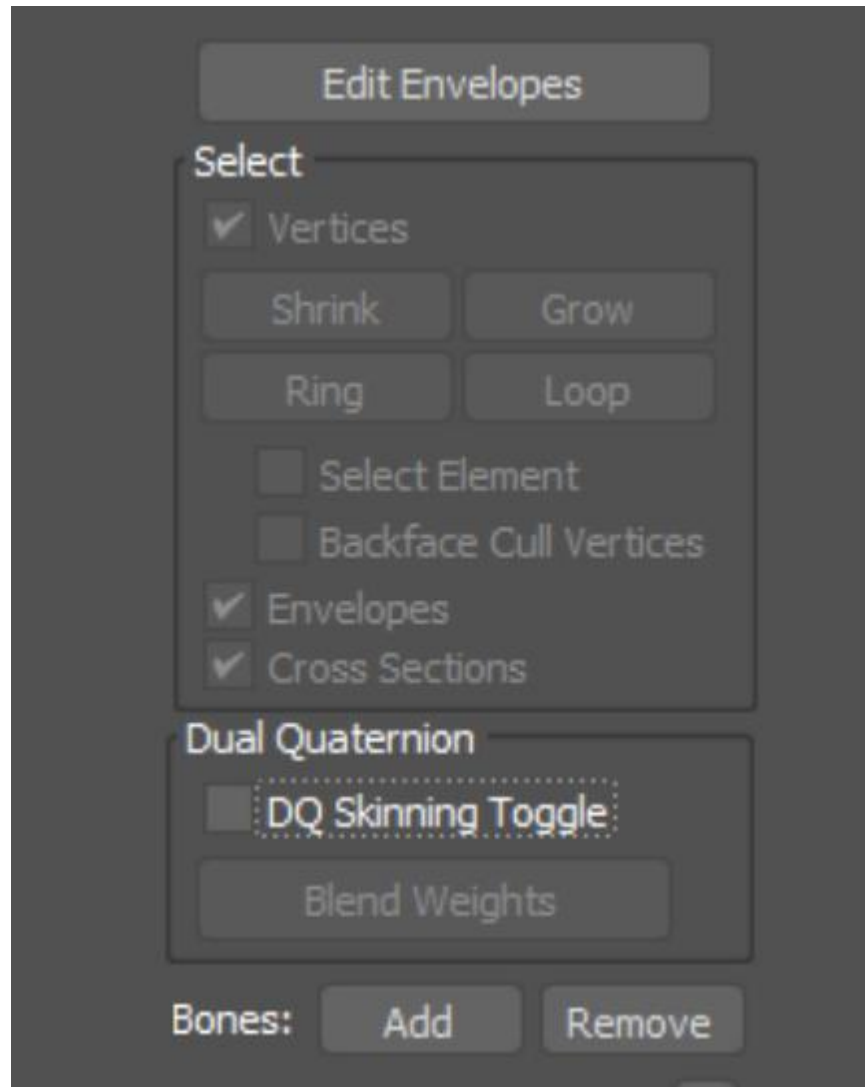
프로젝트에서 가장 실험적인 리깅법을 사용했던 캐릭터입니다.

몸은 이미 죽었으나 영혼이 속박되어 살아있다는 컨셉을 가지고 있는 캐릭터로,
개인적으로 프랑스식 이름인 점이 마음에 들었습니다.

캐릭터의 자체적인 움직임보다는 천이나 피부의 흔들림을 표현하고자 노력했던 기억이 납니다.

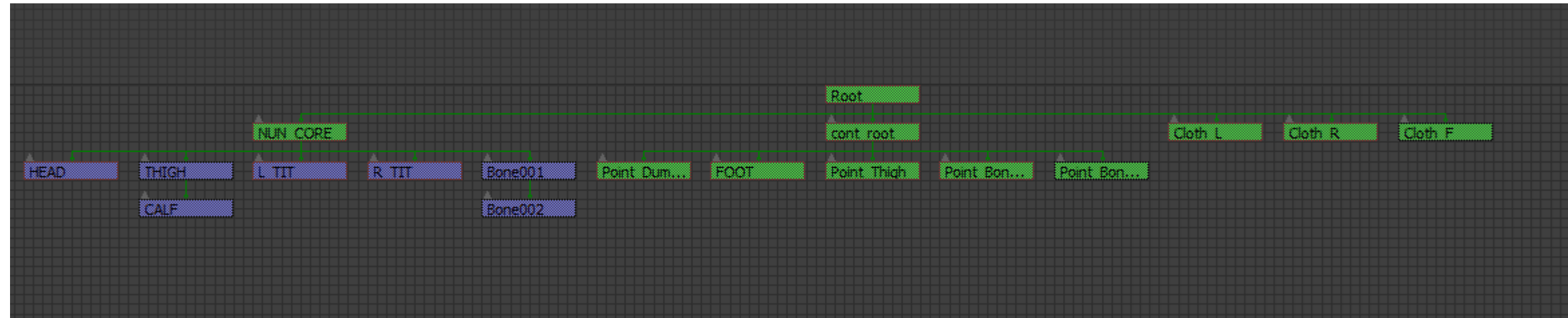
리깅된 캐릭터의 포즈를 잡고 앉히는게 아닌 앉아있는 캐릭터에 bones를 심는 작업이라 고민이 제법 많았습니다.

01. Rigging

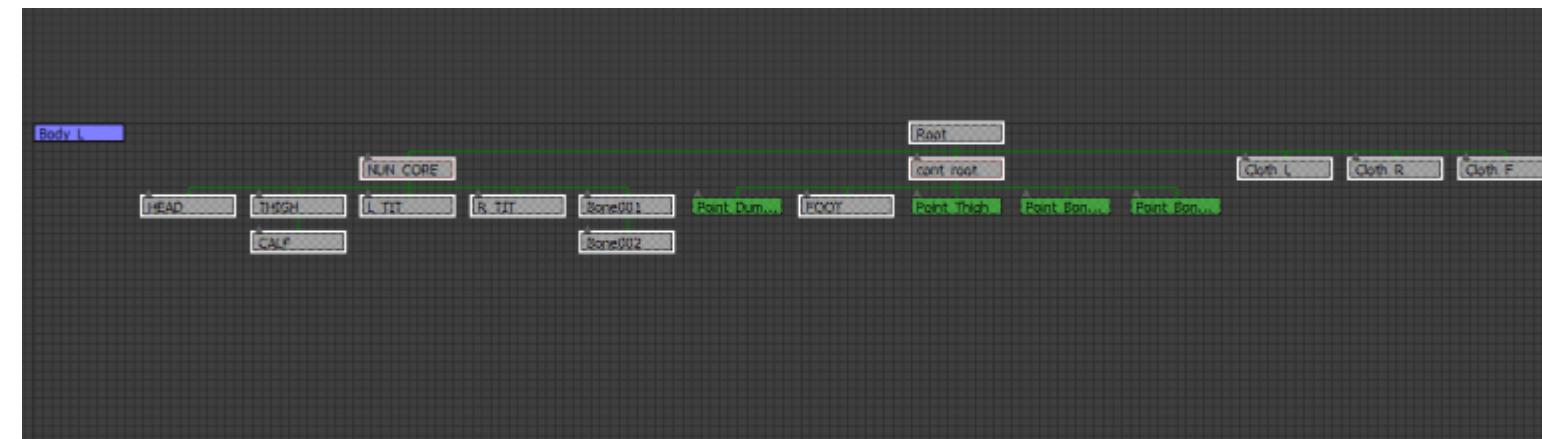


언리얼엔진에 임포트 되는 게임용 리소스임으로 dq를 비활성화 하고 bone affect limit을 6으로 설정하였습니다.

01. Rigging

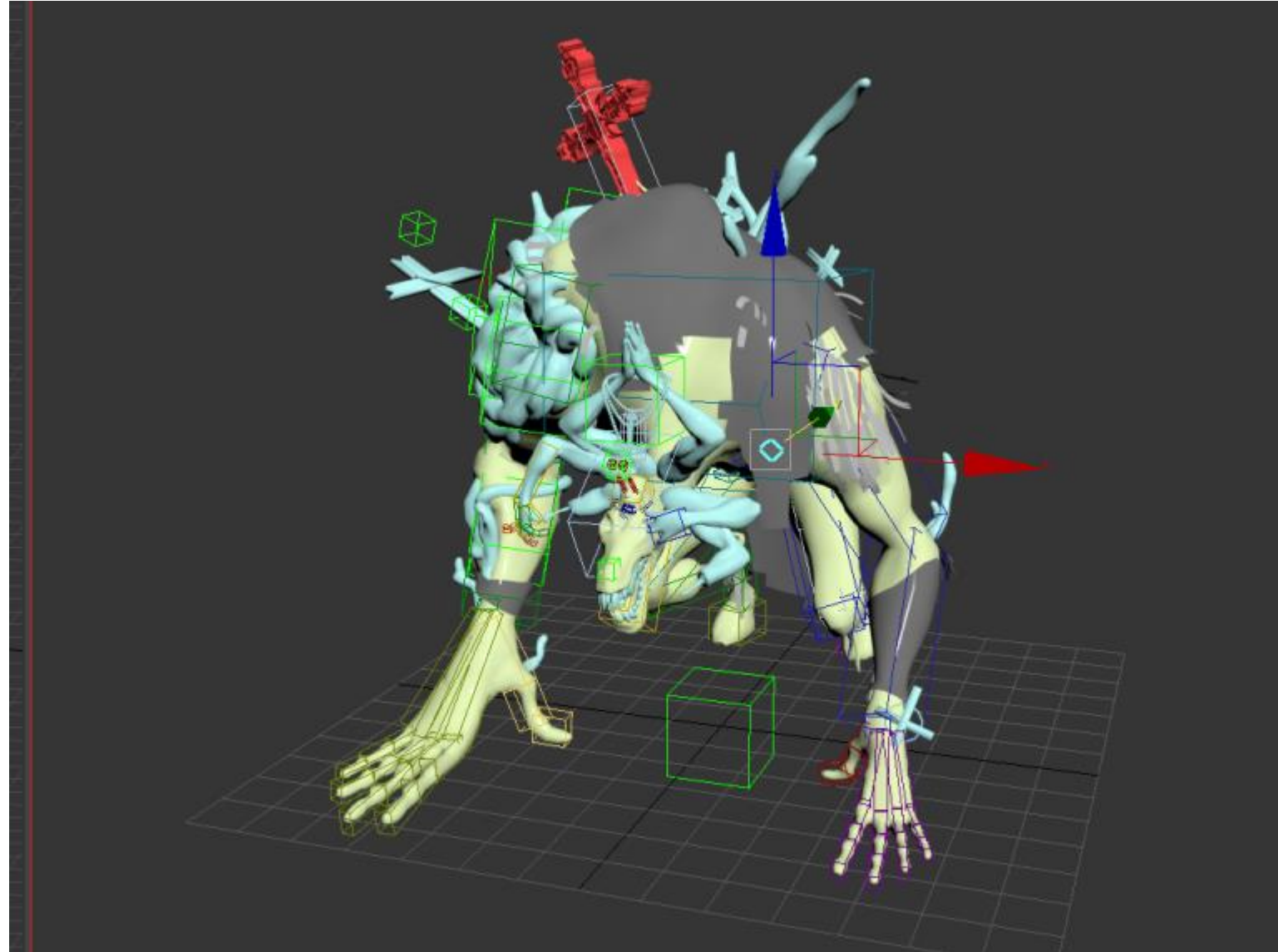


리깅에 사용된 본은다음과 같이 18개를 사용하였습니다.



최종 익스포트 되는 본은 14개 입니다.

01. Rigging



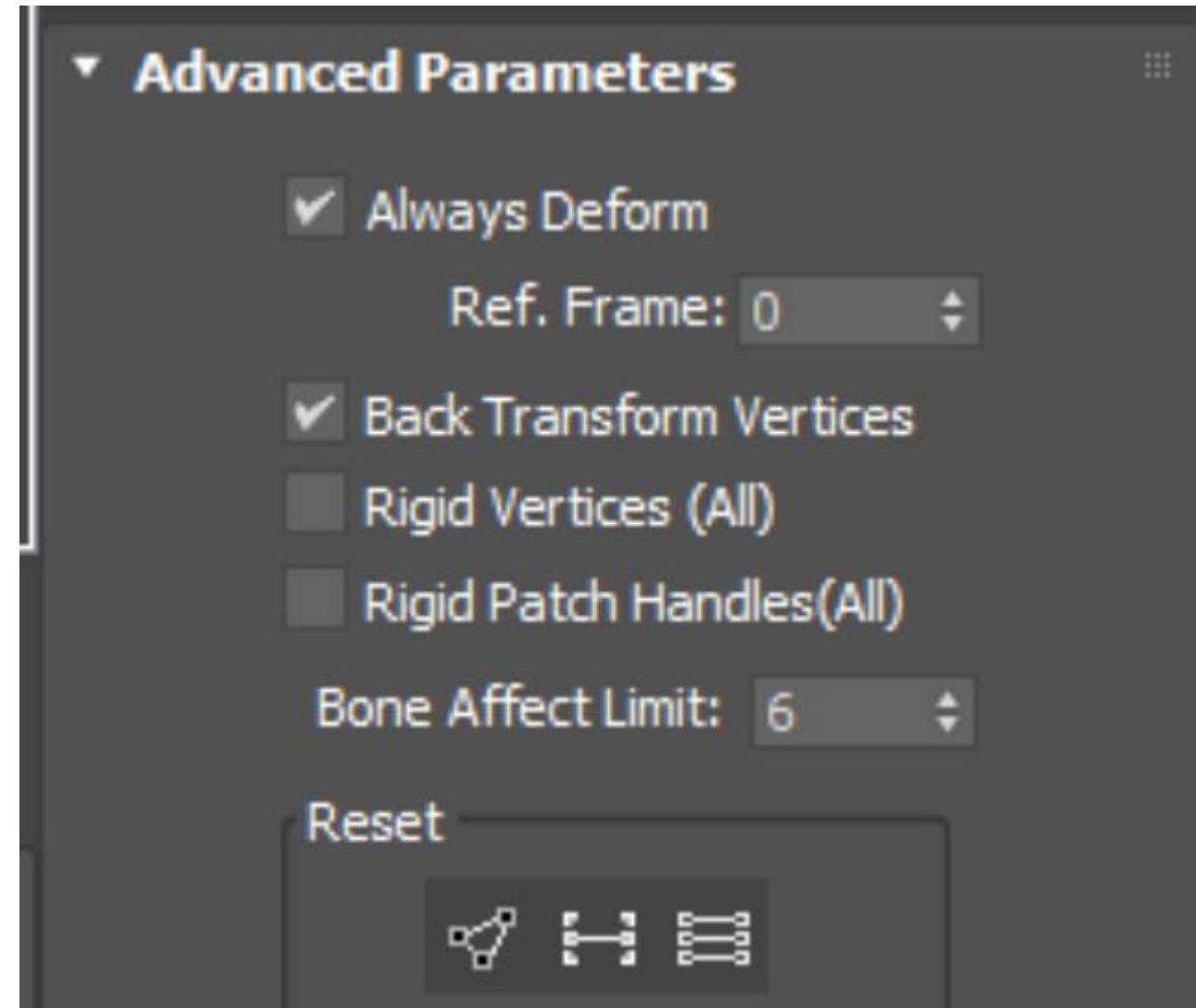
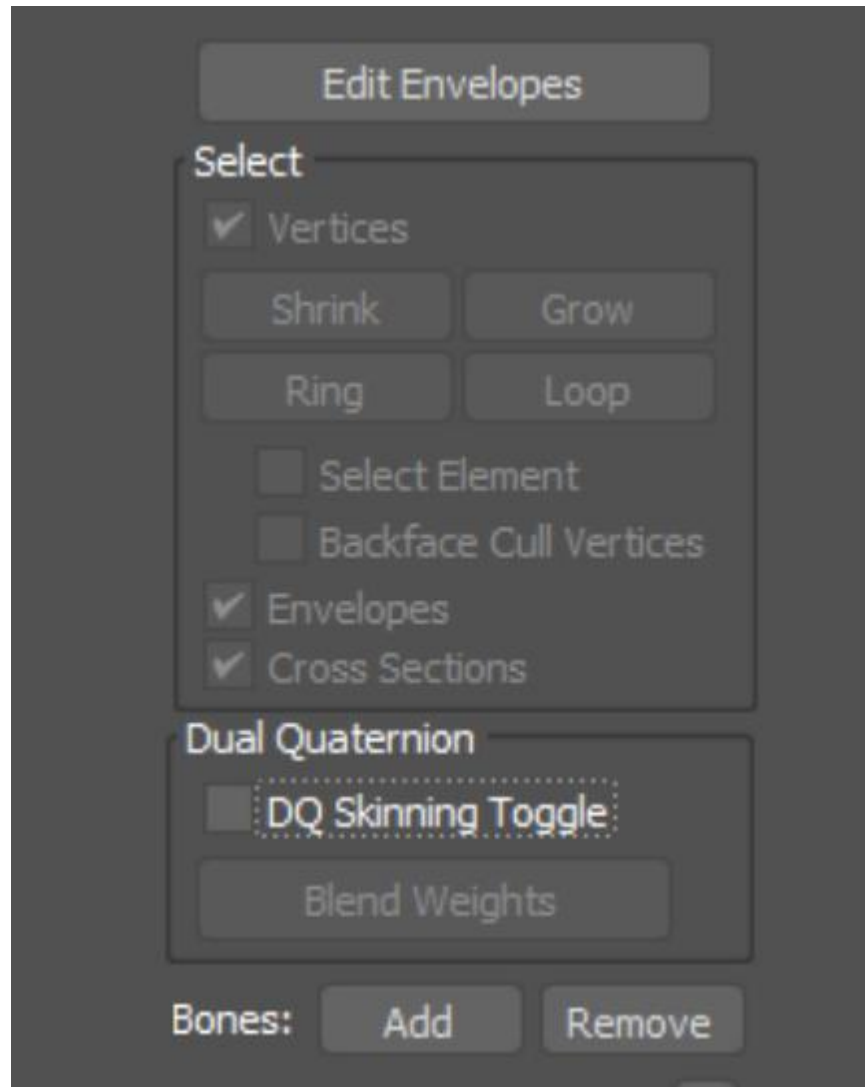
Abel02

어둠에 잠식되어 괴물이 되어버린 성직자 라는 컨셉을 가진 캐릭터로,

쿼터니언회전을 사용하는

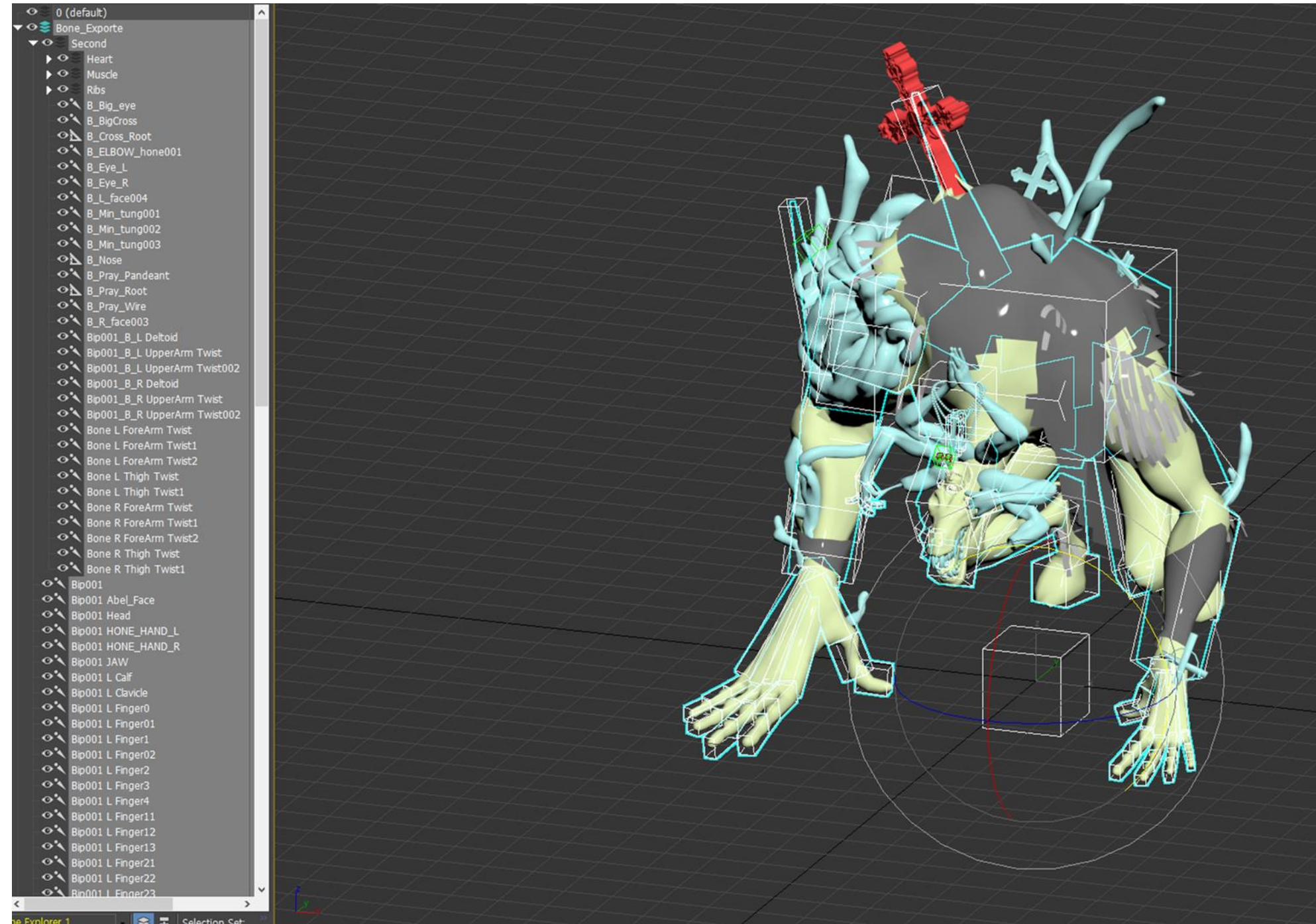
rotation 스크립트로 이루어진 twist bone과 scale animation을 위한 세팅 등 여러가지 새로운 시도를 많이 해본 작업물입니다.

01. Rigging



언리얼엔진에 임포트 되는 게임용 리소스임으로 dq를 비활성화 하고 bone affect limit을 6으로 설정하였습니다.

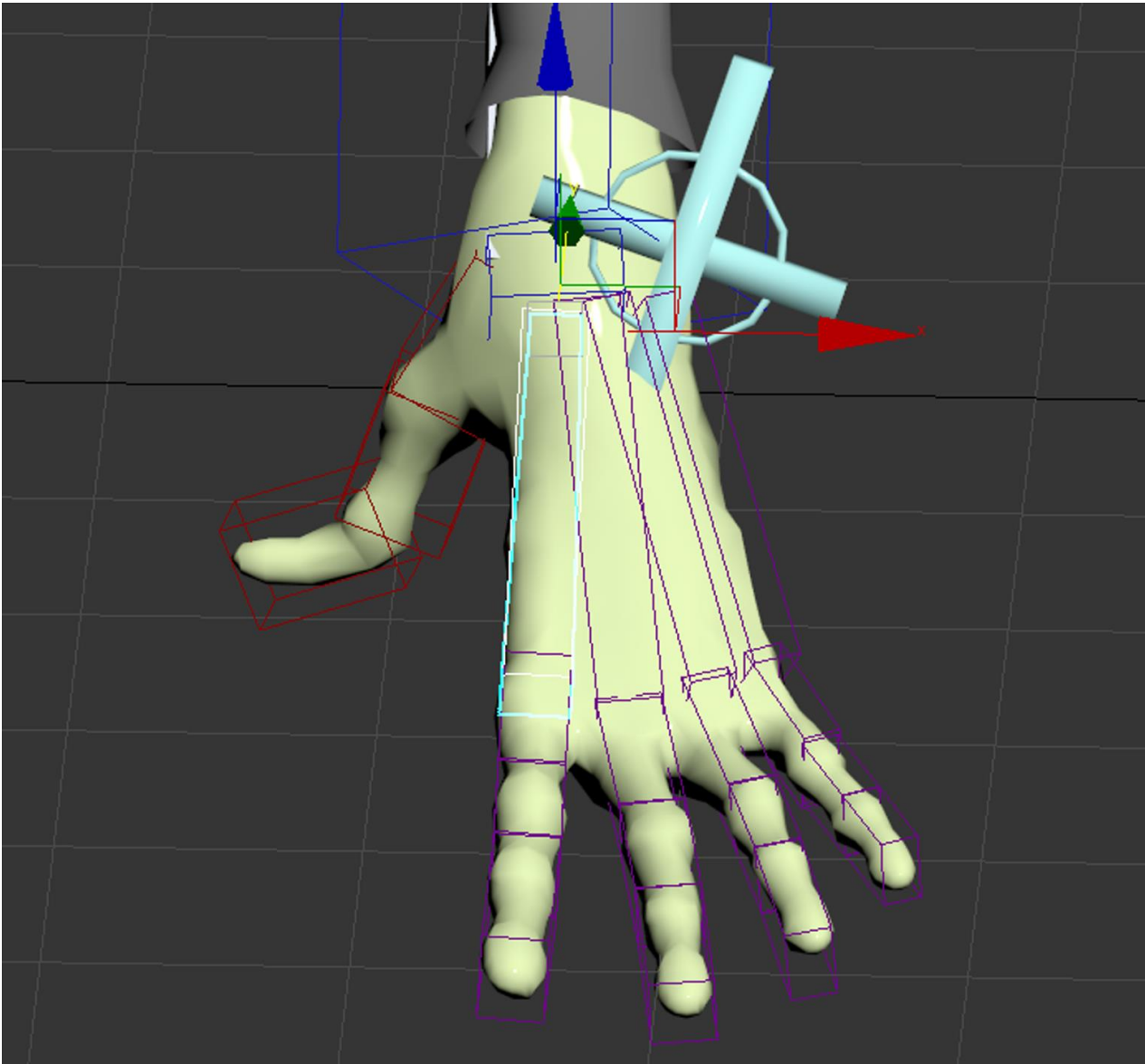
01. Rigging



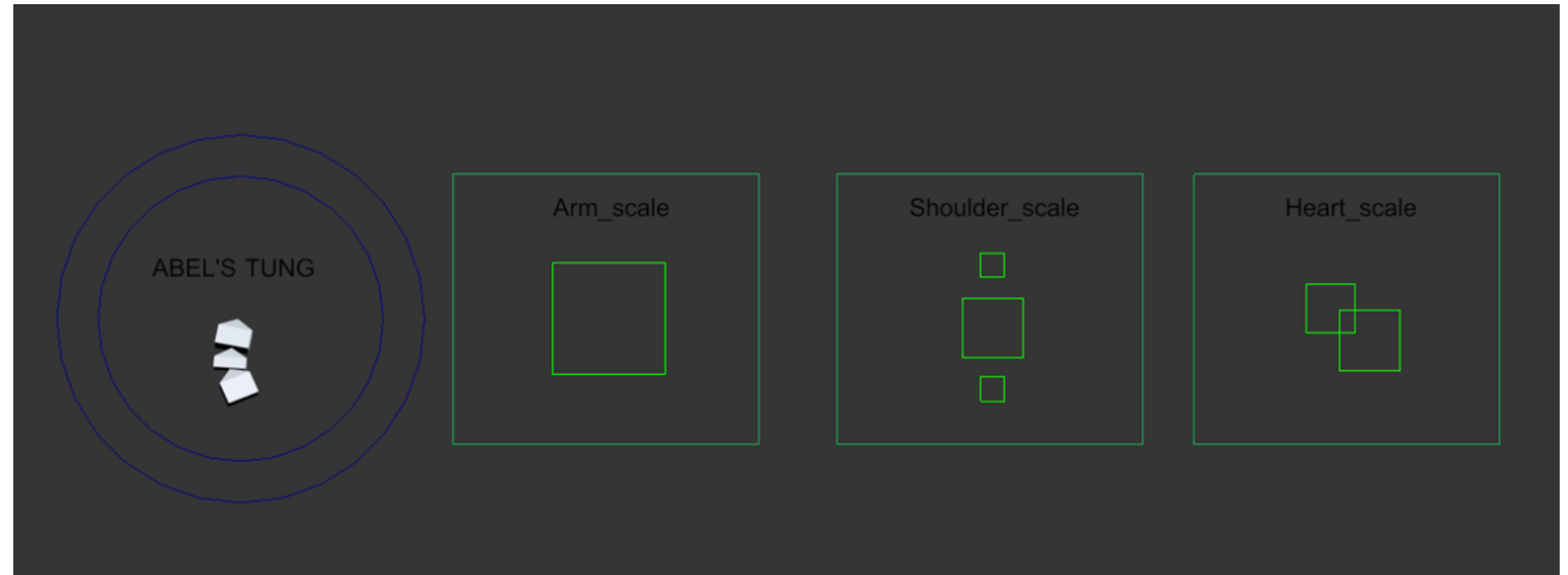
당시 3ds max로 애니메이션을 작업하는 컴퓨터 사양을 고려하여 스키닝용 본을 따로 만들지않고

익스포트에 필요한 본들을 한쪽 레이어에 몰아 불필요한 헬퍼나 nub을 익스포트 되지 않게 하였습니다.

01. Rigging



화면에서 크게보이는 몬스터였기 때문에 손가락의 세팅을 보다 리얼한 방식으로 작업했습니다.

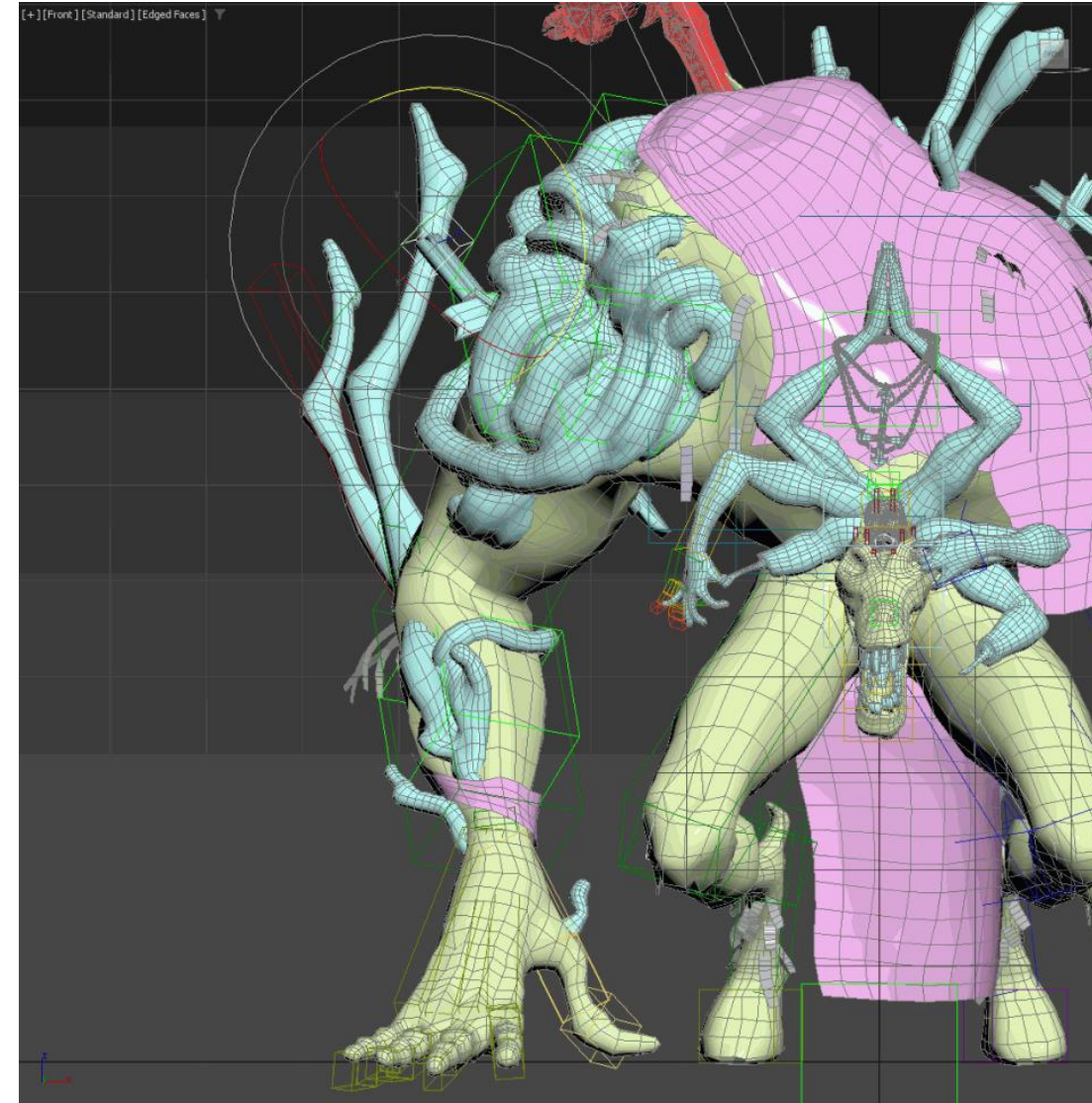


크리쳐 이마 속 작은 얼굴의 혀를 컨트롤 하기위한 컨트롤러를 orientation constraint 로 연결하고
각부위의 스케일 애니메이션을 컨트롤 하는 컨트롤러를 Wire parameter로 제작하였으나
심장을 제외하곤 컨트롤러를 연결한 본들이 외부에서도 눈에 잘 띄어 제작한 컨트롤러들이
불필요하다고 판단되어 과감히 삭제하였습니다.

01. Rigging



orientation constraint와 position constraint 컨트롤러를 사용하여 팔에 박힌 십자가를
컨트롤 할수있도록 세팅하였습니다.



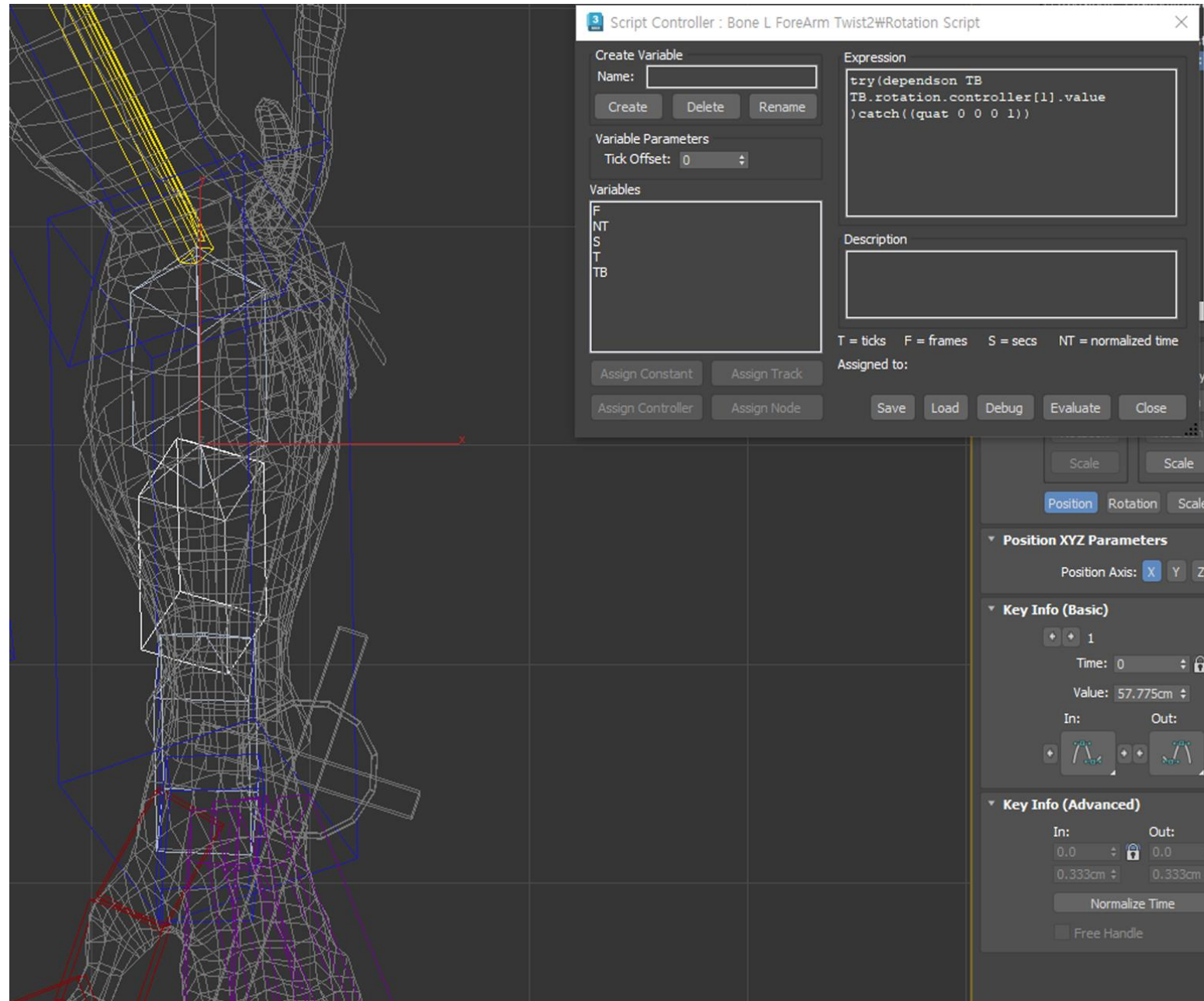
큰 허의 경우 posture copy를 자주 사용하게 될것 같아 바이패드의 xtra본을 사용 하였습니다.

코의 경우 스케일 애니메이션을 주기 위해 point Helper로 리깅 하였고

스케일, 로테이션 ,포지션을 모두 사용할 수 있습니다.

손 모양의 뿔 또한 xtra 본을 활용하였습니다.

01. Rigging

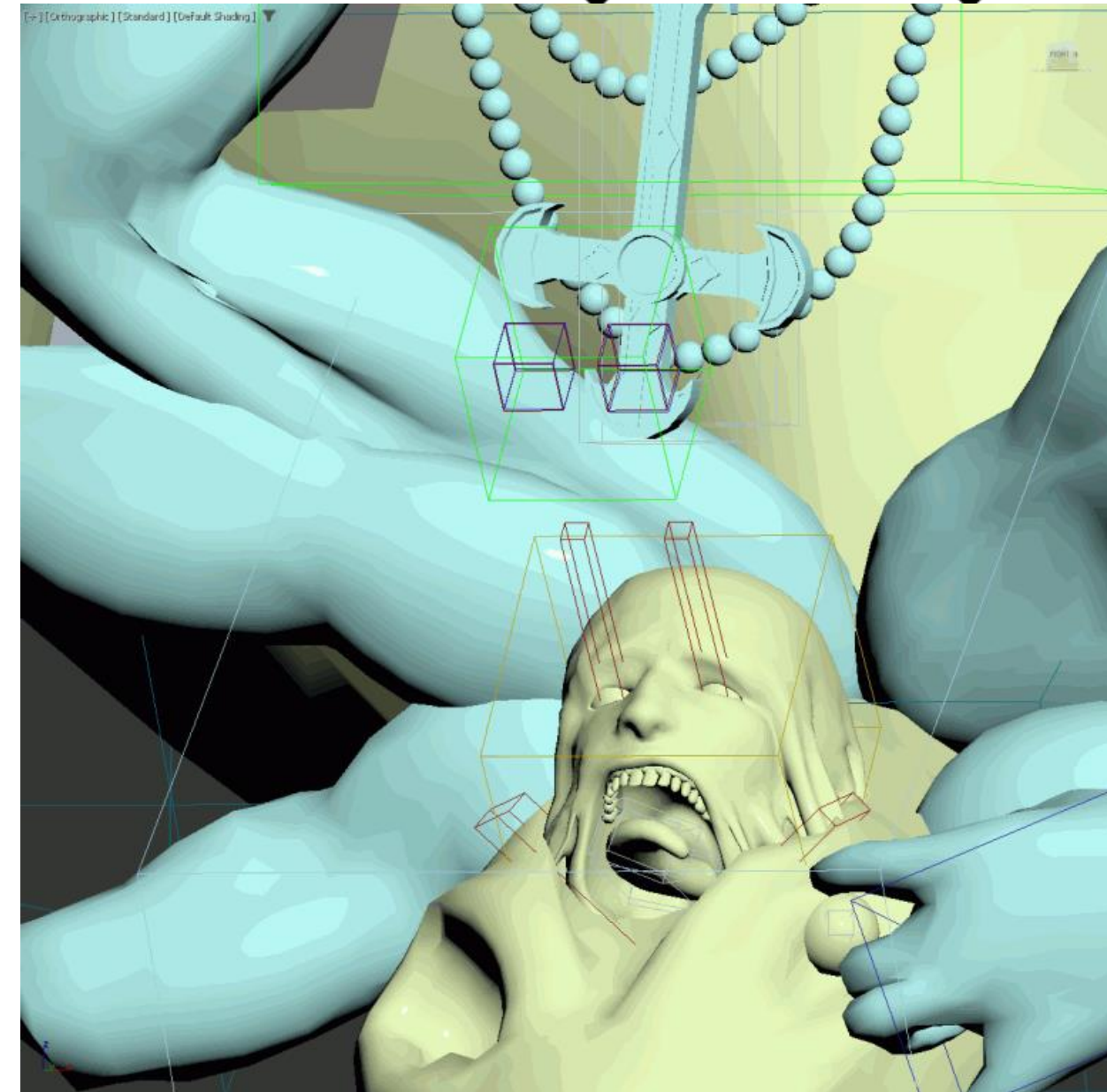


하박에 트위스트본을 설치하였습니다 .

통상시엔 look at constraint 와 orientation constraint를 활용한 본 세팅을 직접 했지만

해당 작업을 할땐

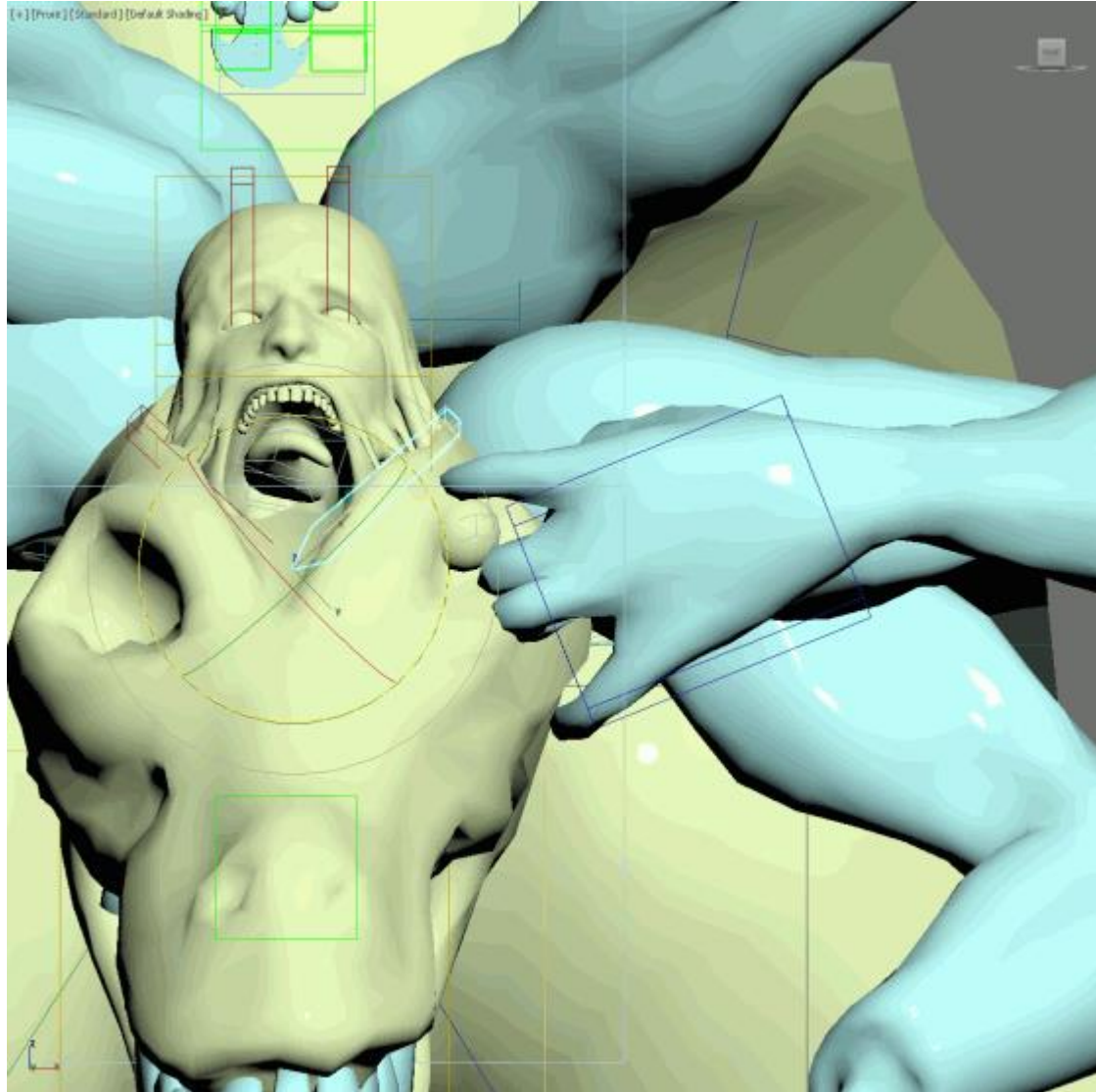
felix 쿼터니언 회전법을 사용하는 스크립트를 사용하여 작업 해보았습니다.



안구의 경우 두 눈을 동시에 컨트롤 할 수 있는 컨트롤러를 만들었고

두 안구가 각각 따로 움직이게도 가능하게 설계 하였습니다.

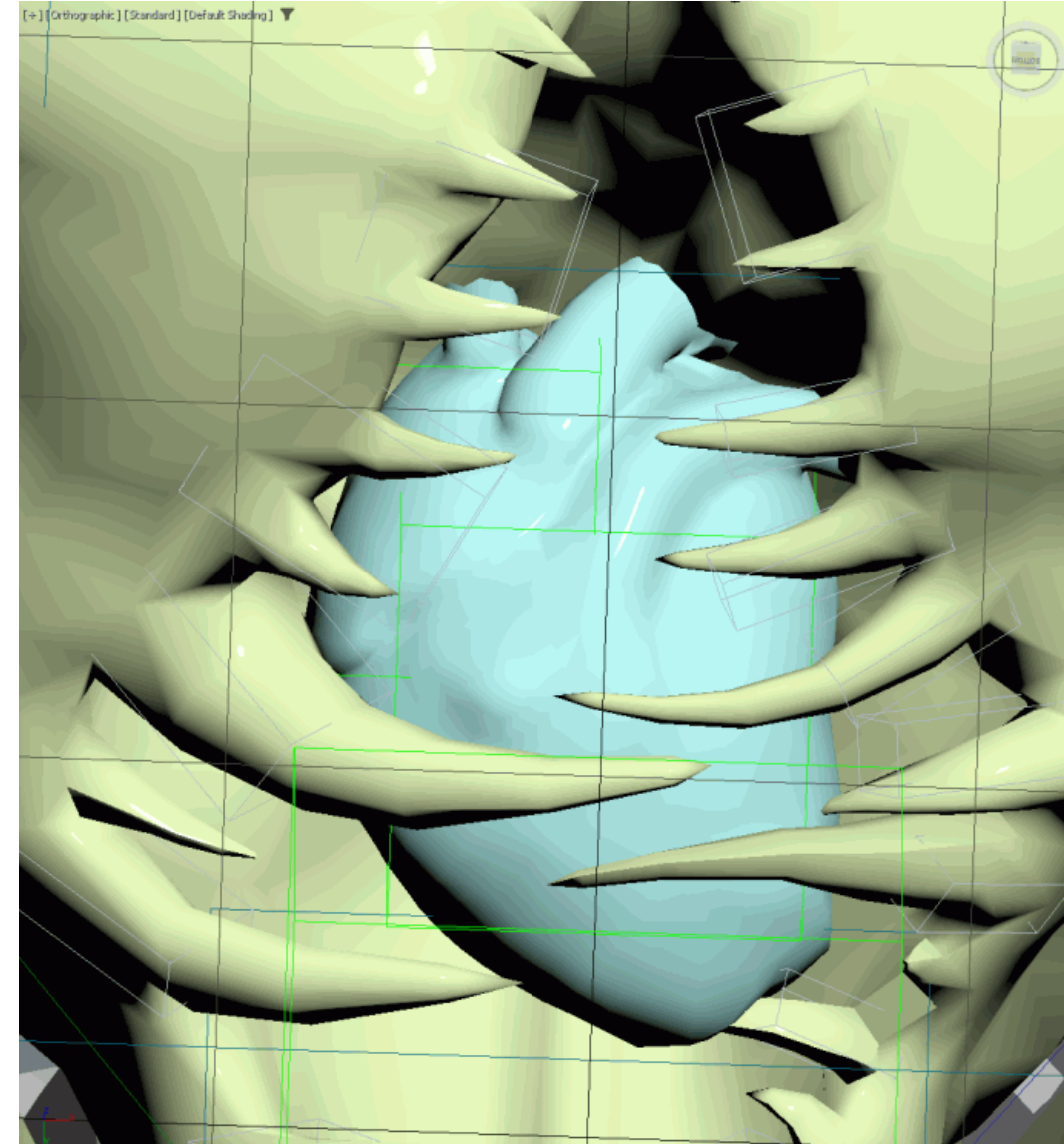
01. Rigging



파묻힌 얼굴을 덮을 수 있게끔 본을 세팅했습니다.

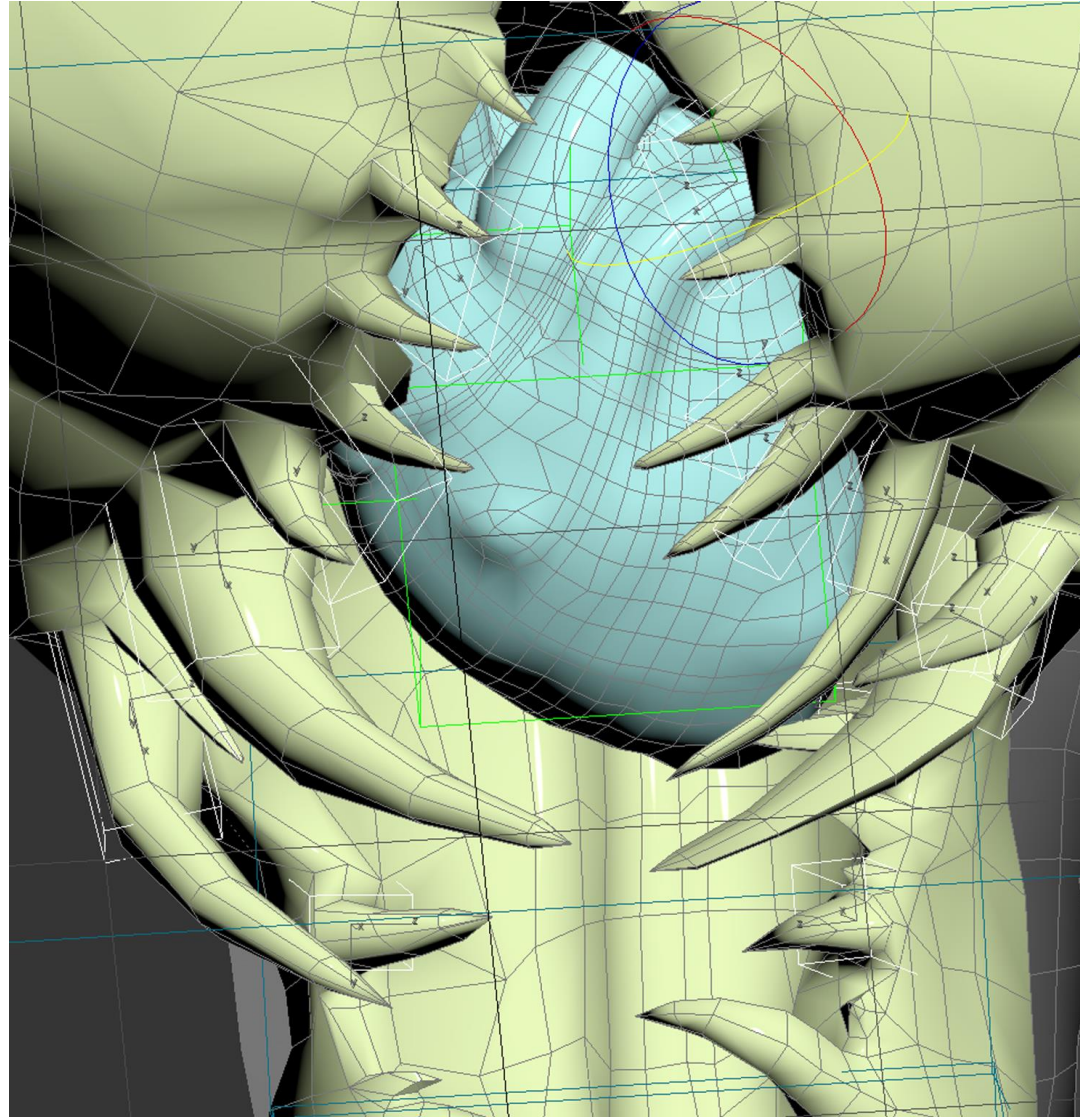
해당 움직임을 상정하고 만들어진 모델링이 아니라 폴리곤 구조가 다소 아쉬웠지만

이런 동작도 있으면 재미있을 것 같다는 생각에 최소한으로 본을 설치해서 구현하였습니다.

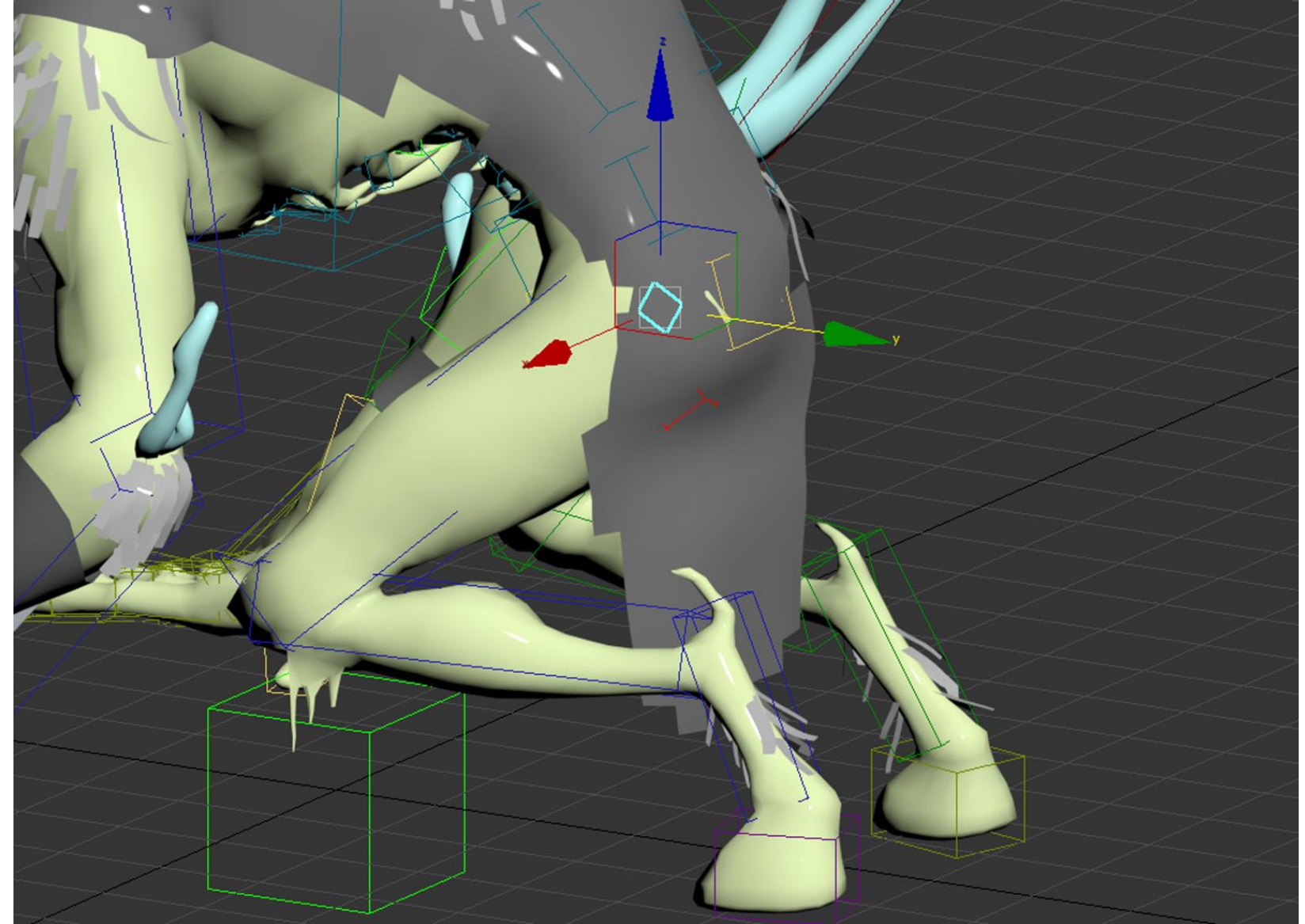


혈관과 마찬가지로 심장의 움직임을 헬퍼 2개로 구현하였습니다.

01. Rigging



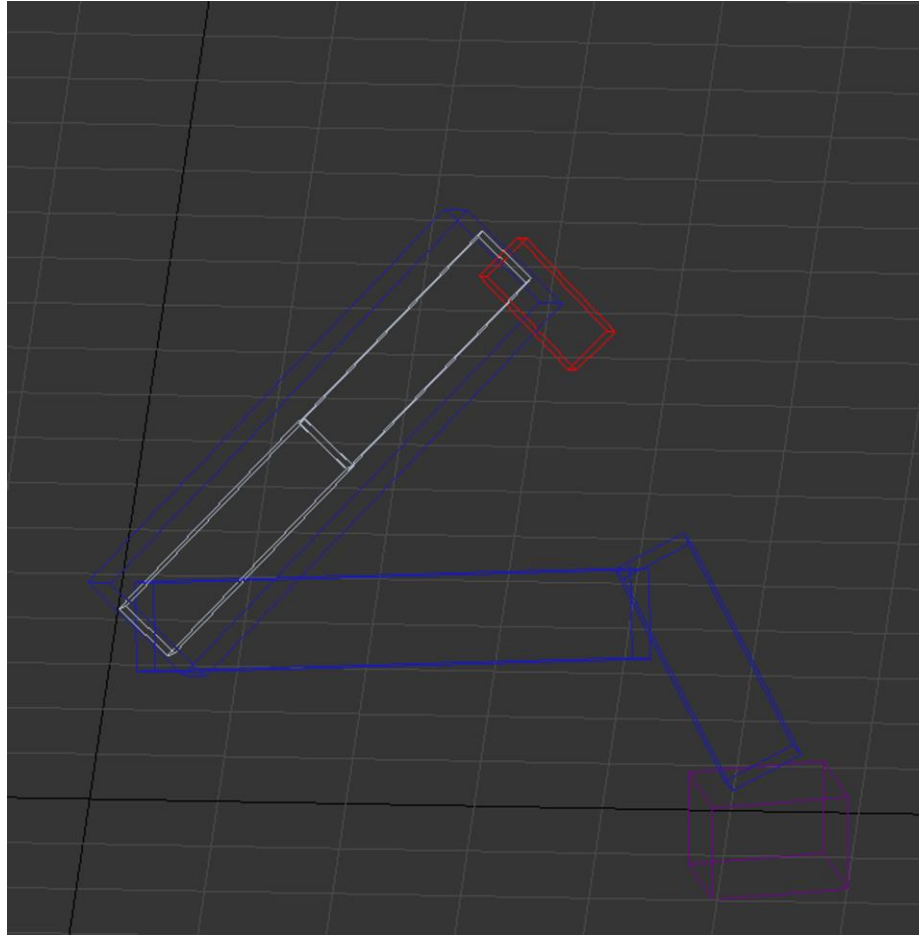
그로테스크함을 강조하고자 갈비뼈가 움직일수 있게 각 본들을 설치 하였습니다.



천의 경우 UE5의 CLOTH 기능을 사용하기 때문에

별도의 본세팅을 하지 않았습니다.

01. Rigging



애니메이션 작업시에 큰 고민을 안겨주었던 역관절 세팅이었습니다.

TOE를 FOOT으로 사용하는 세팅을 하였다가

이후 IK포인트를 지정하는데 있어 불편함이 컸었습니다.

그러나 당시 개발기한이 촉박 했기에 리깅을 수정하고 다시 애니메이션 리소스를 만들 순 없었기에 그대로 진행하였습니다.

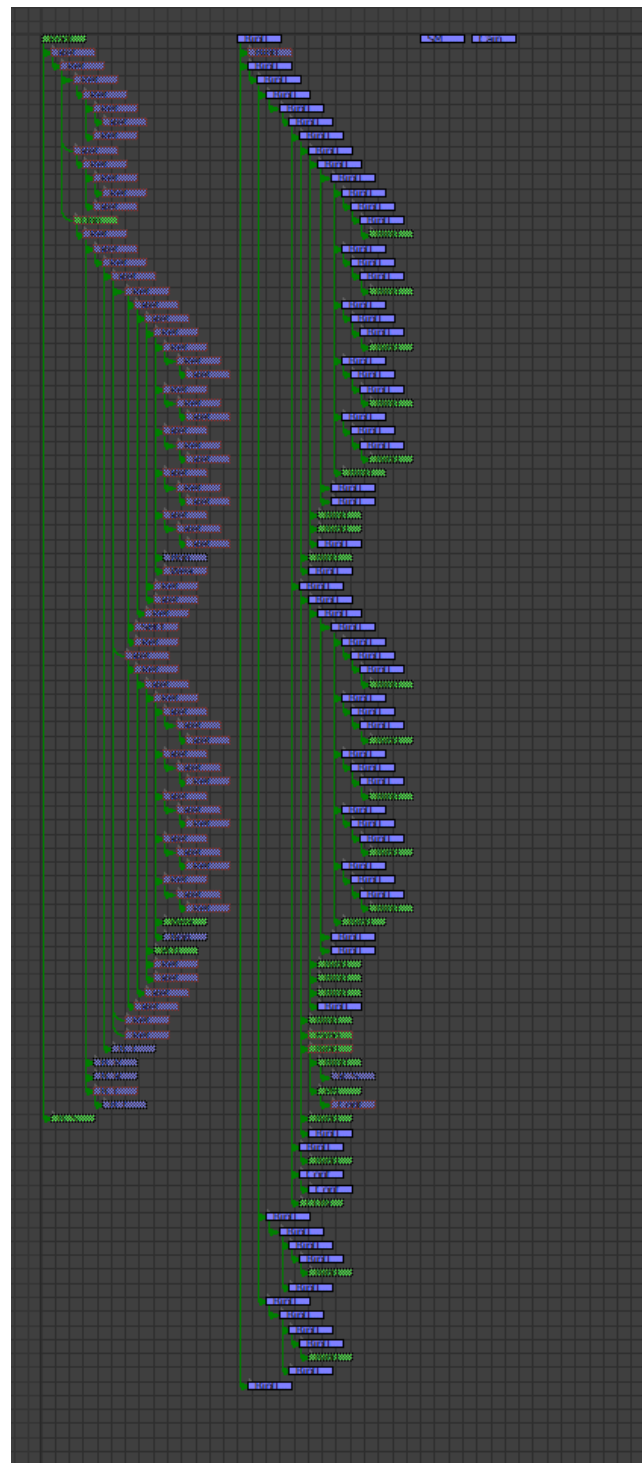
01. Rigging



Player Charactor

인게임에서 등장하는 주인공 캐릭터 입니다.
망토와 치마의 경우 UE5의 Cloth 기능으로 구현 하였으며
실험적인 세팅을 많이 해볼 수 있었기 때문에 제 성장의 밑거름이 되어준 캐릭터 입니다.

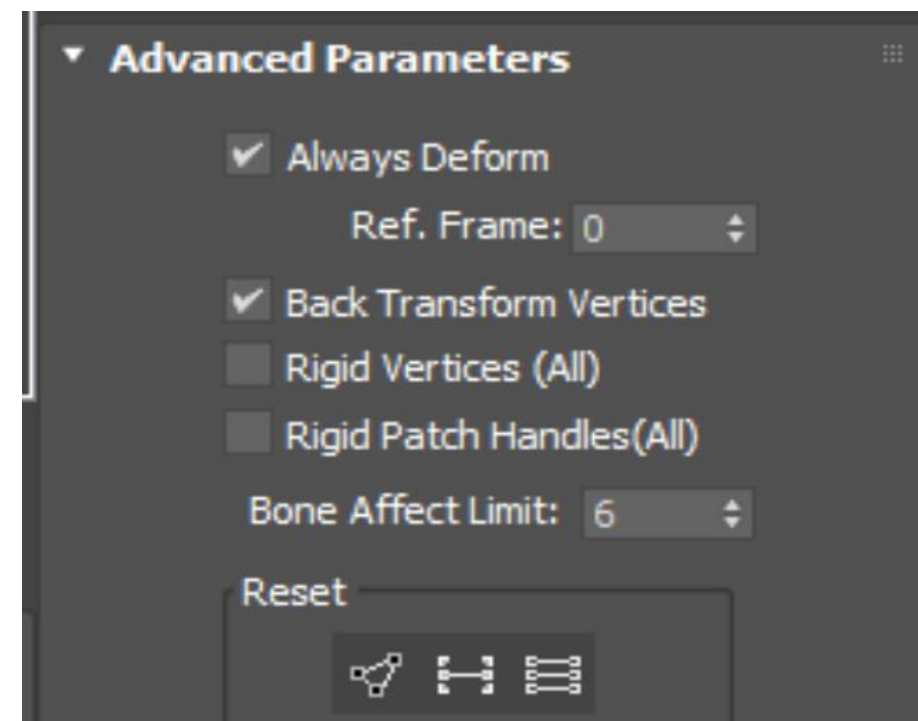
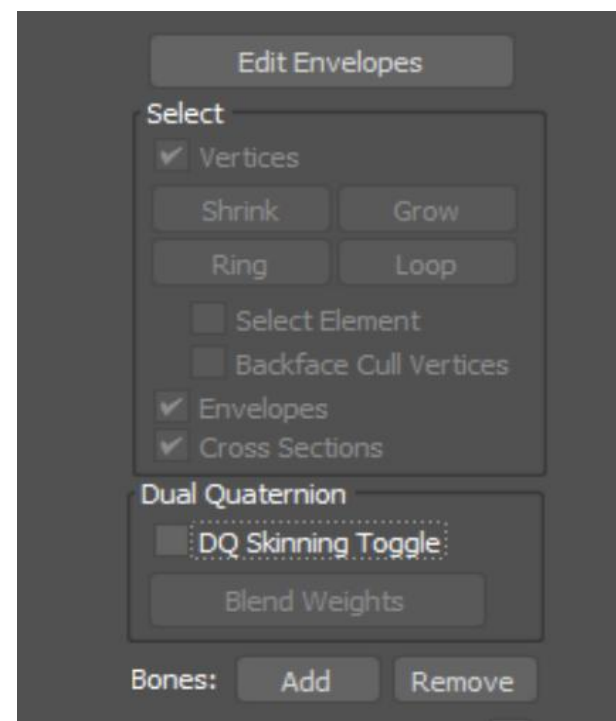
01. Rigging



bipad 를 기본 뼈대로 하였고

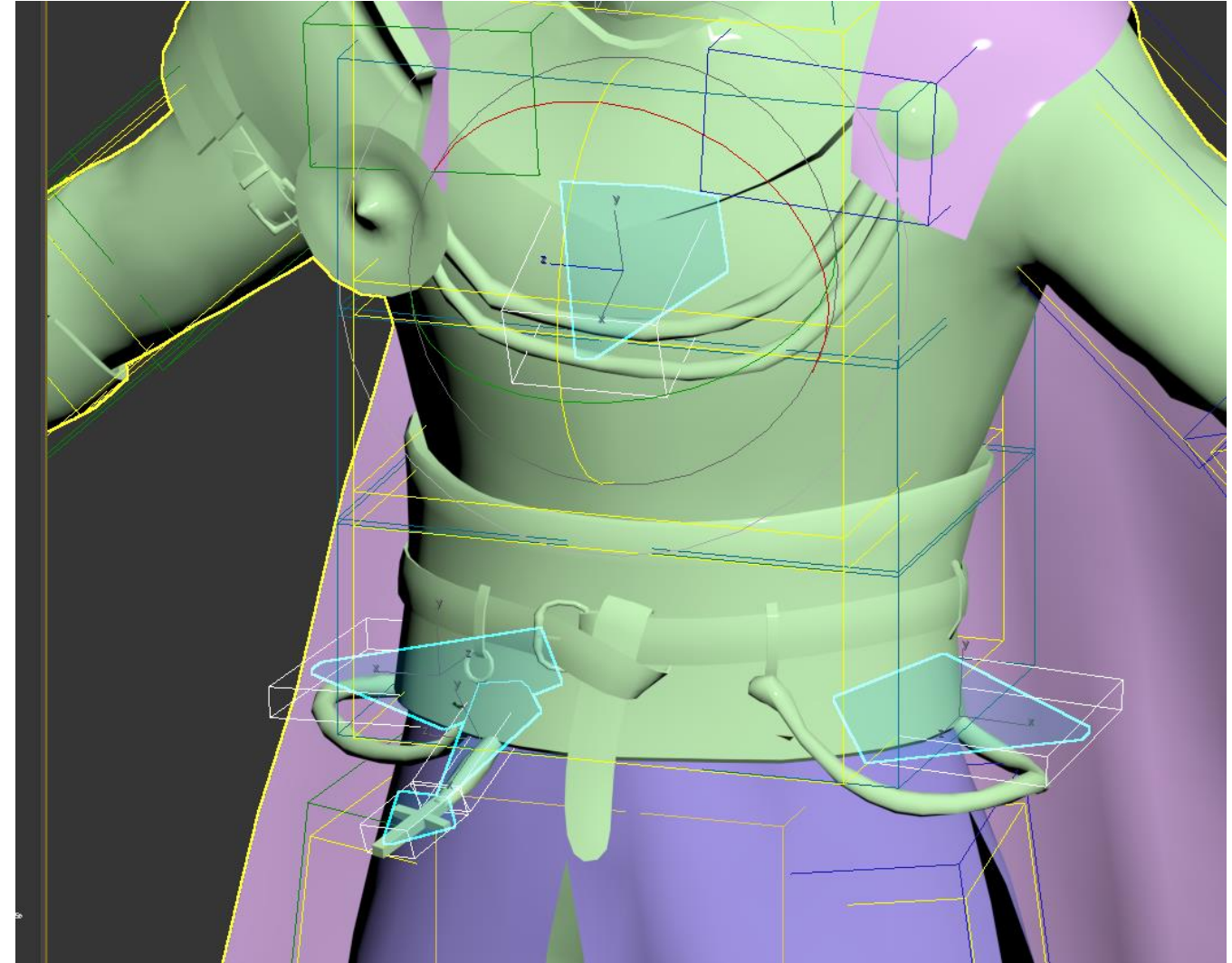
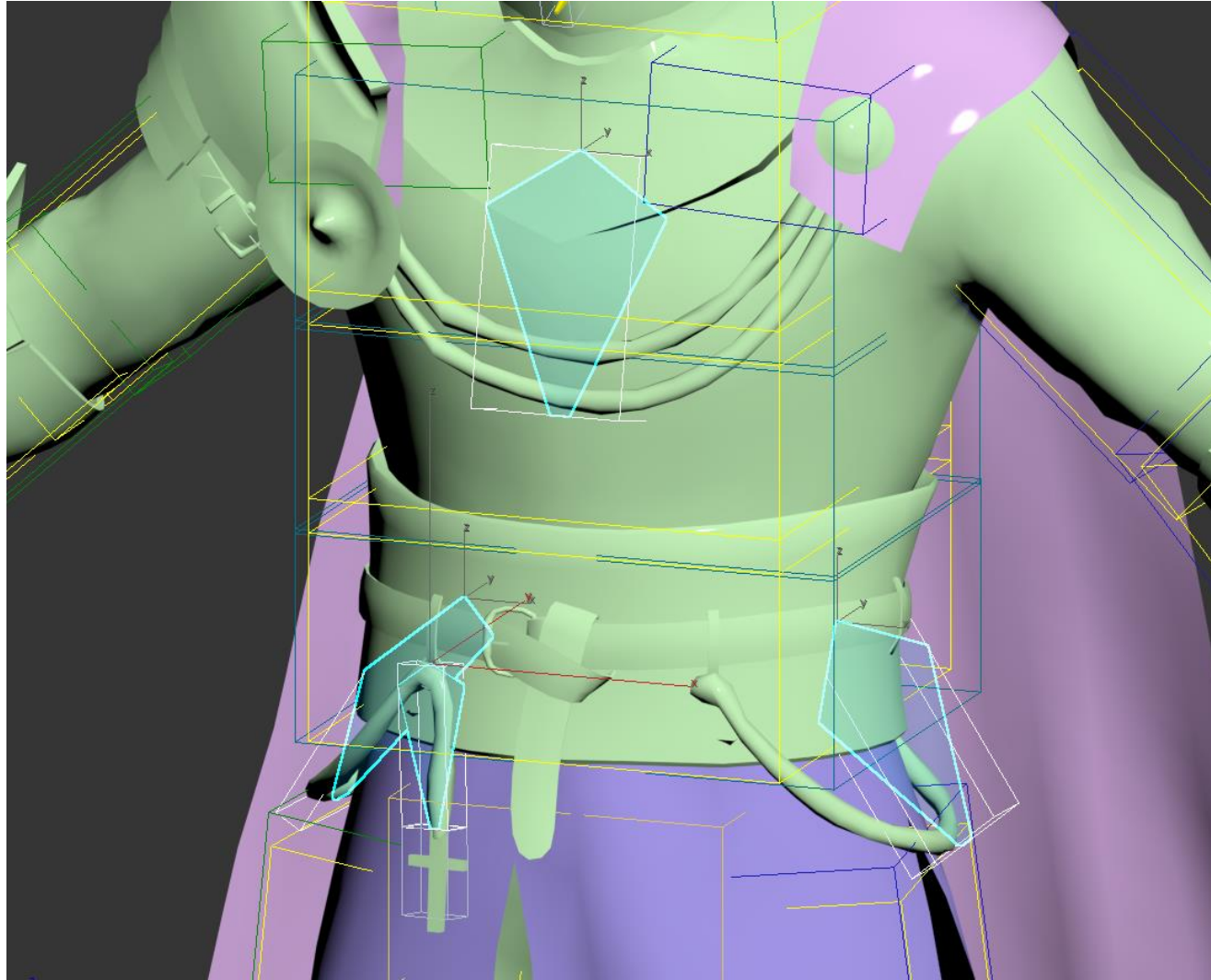
다음과 같이 계층구조및 bone 갯수의 최적화를 위해

원본 BONE의 움직임을 참조하는스키닝, импорт용 BONE을 생성하여 최적화 작업을 하였습니다.



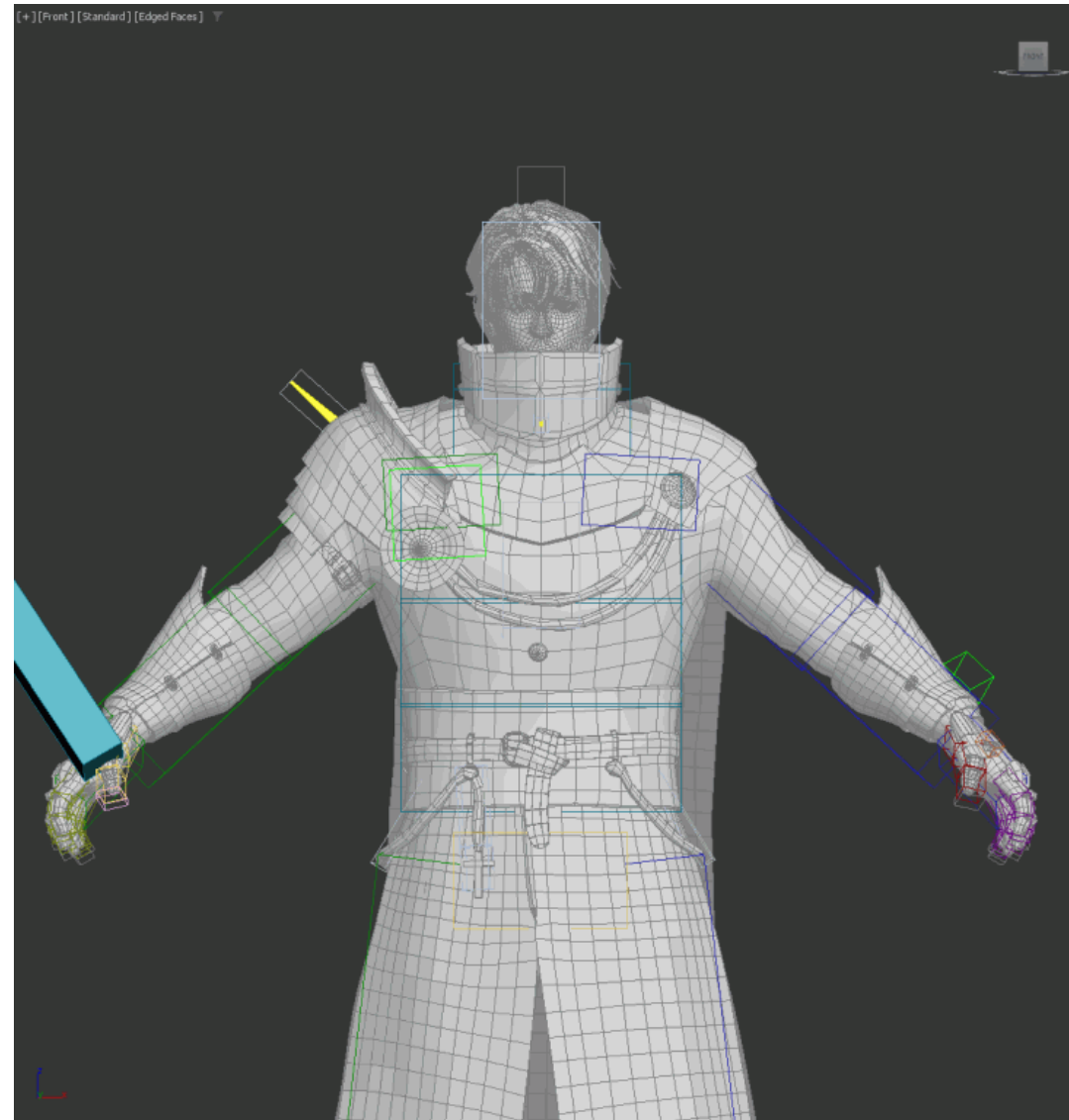
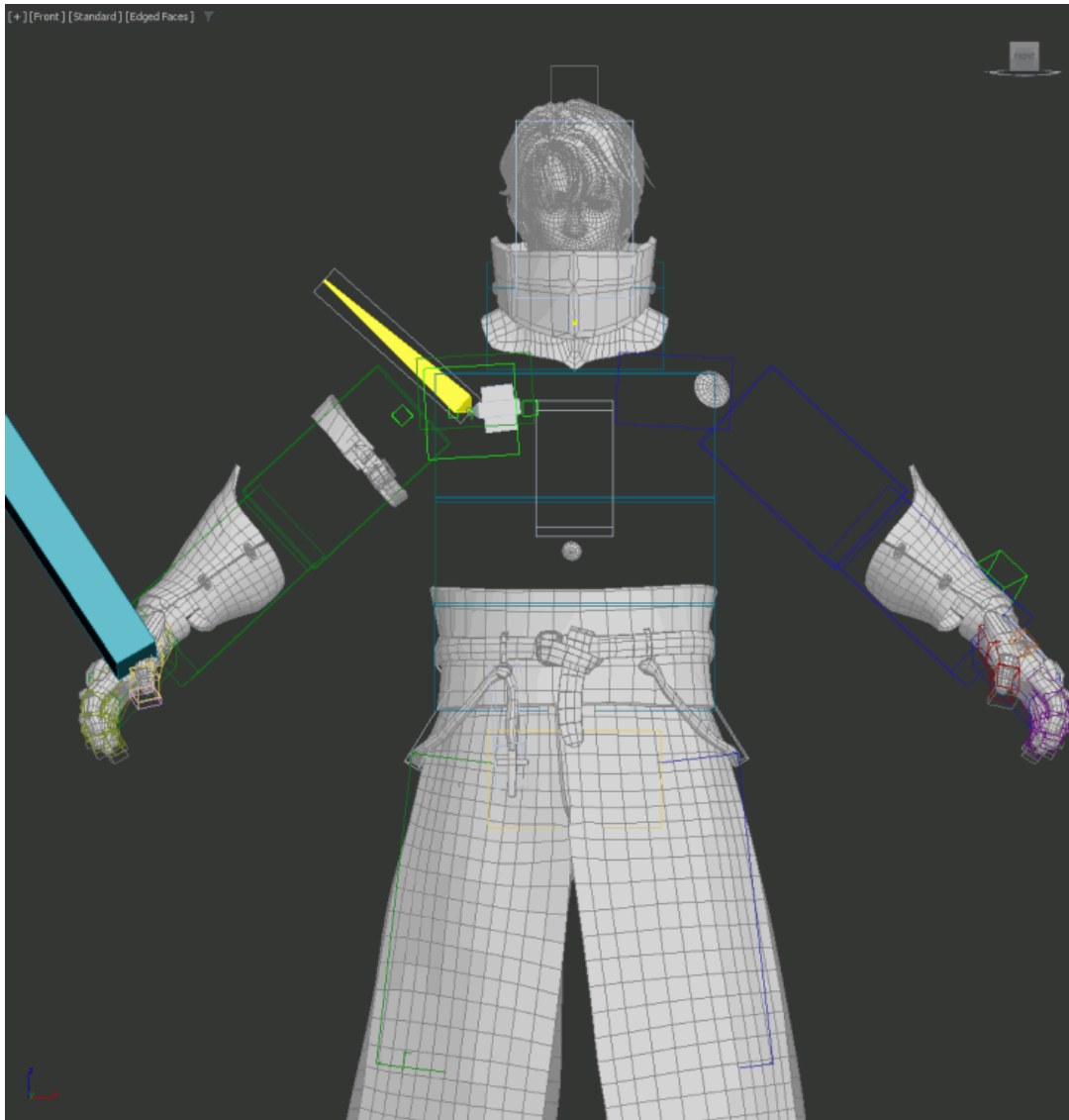
엔리얼엔진에 импорт 되는 게임용 리소스임으로 dq를 비활성화 하고 bone affect limit을 6으로 설정하였습니다.

01. Rigging



간단한 세컨더리의 경우 일반 본으로 리깅하였습니다.

01. Rigging

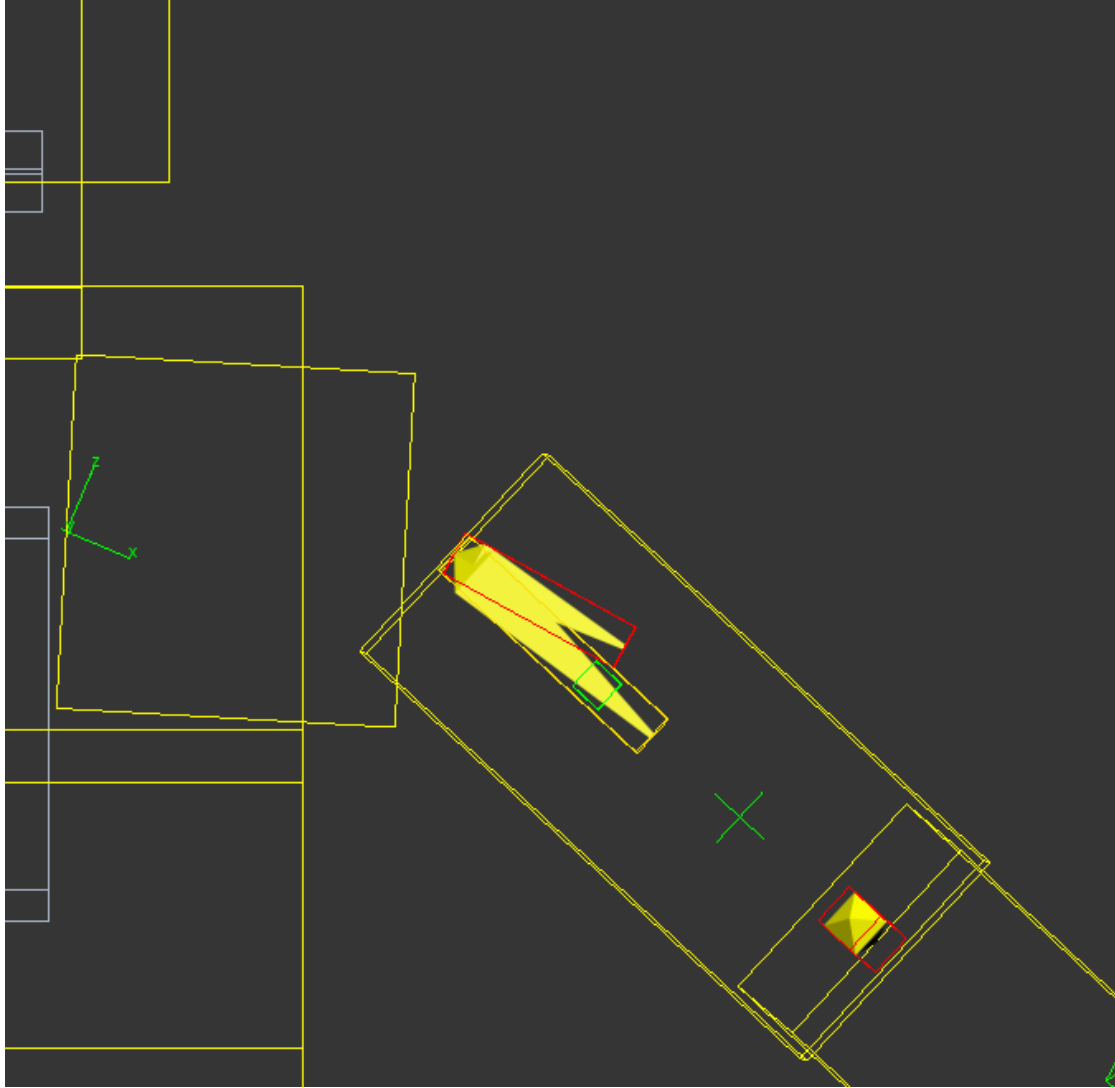


expose TM 헬퍼를 사용하여 상박의 회전값에 따라 본에 position 값에 변화를 주어 어깨갑옷이 어느정도 자동으로 자리를 잡게 세팅하였고

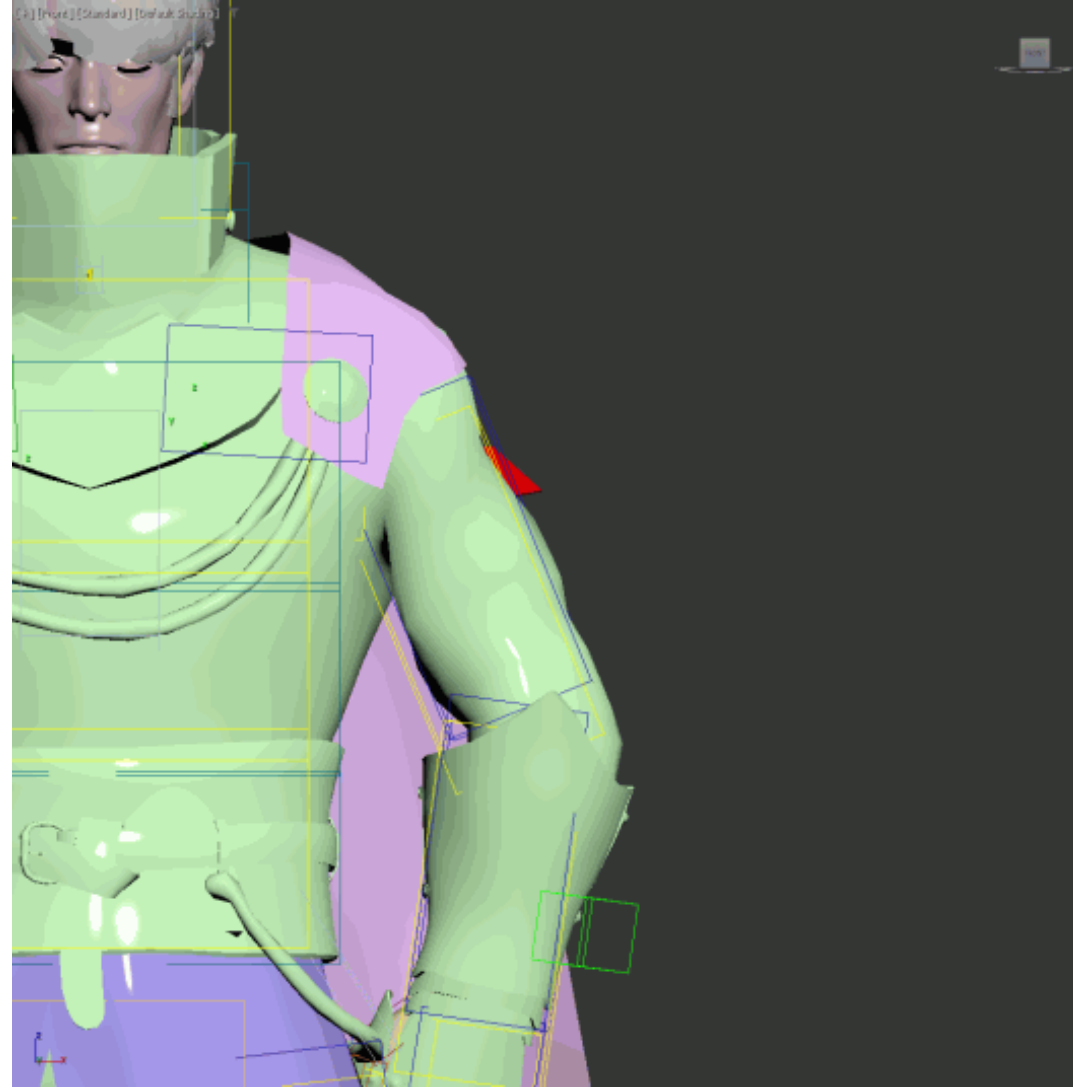
해당본을 자식으로 하는 컨트롤용 본을 만든 뒤

해당 컨트롤러의 포지션과 로테이션 값을 참조하는 익스포트전용 본을 ref 스크립트를 사용하여 만들었습니다.

01. Rigging

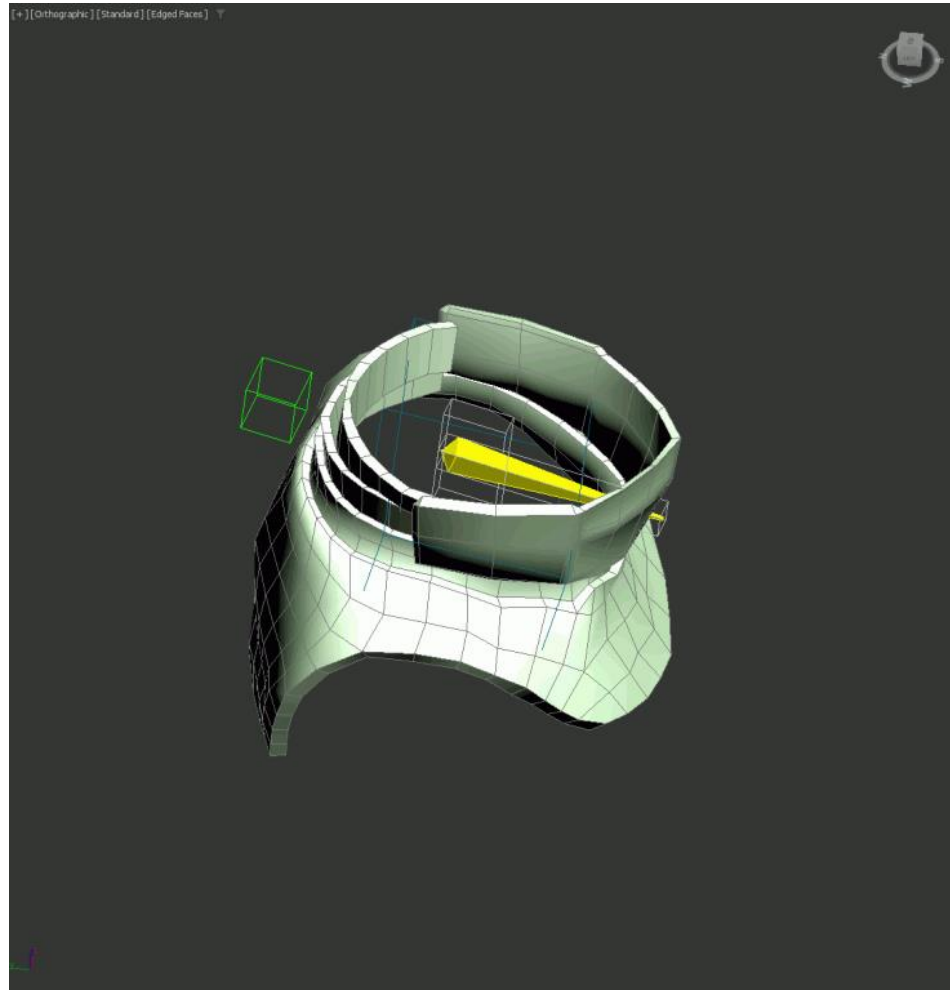


보다 자연스러운 스킨링을 위해
non flip 세팅법을 응용하여 삼각근, 팔꿈치본 등을 적용 하였습니다.



삼각근의 결과물은 다음과 같습니다.

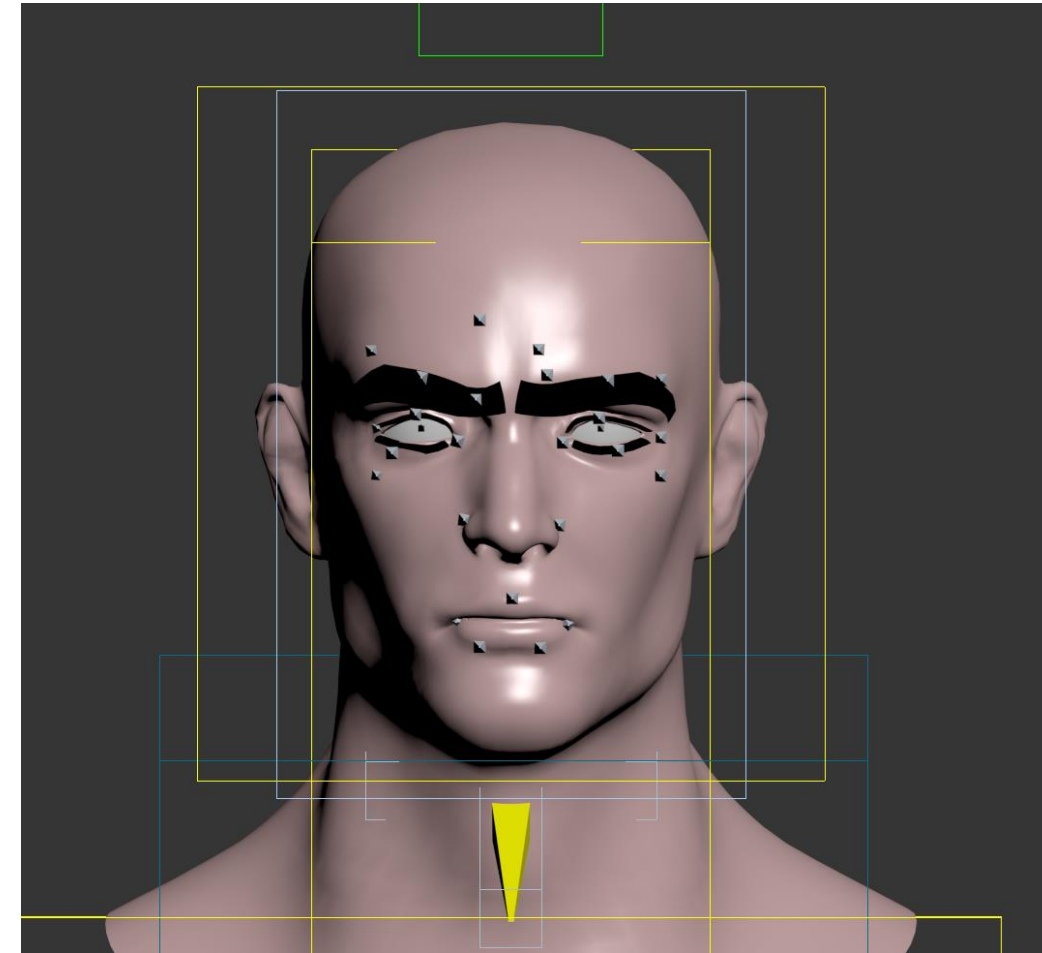
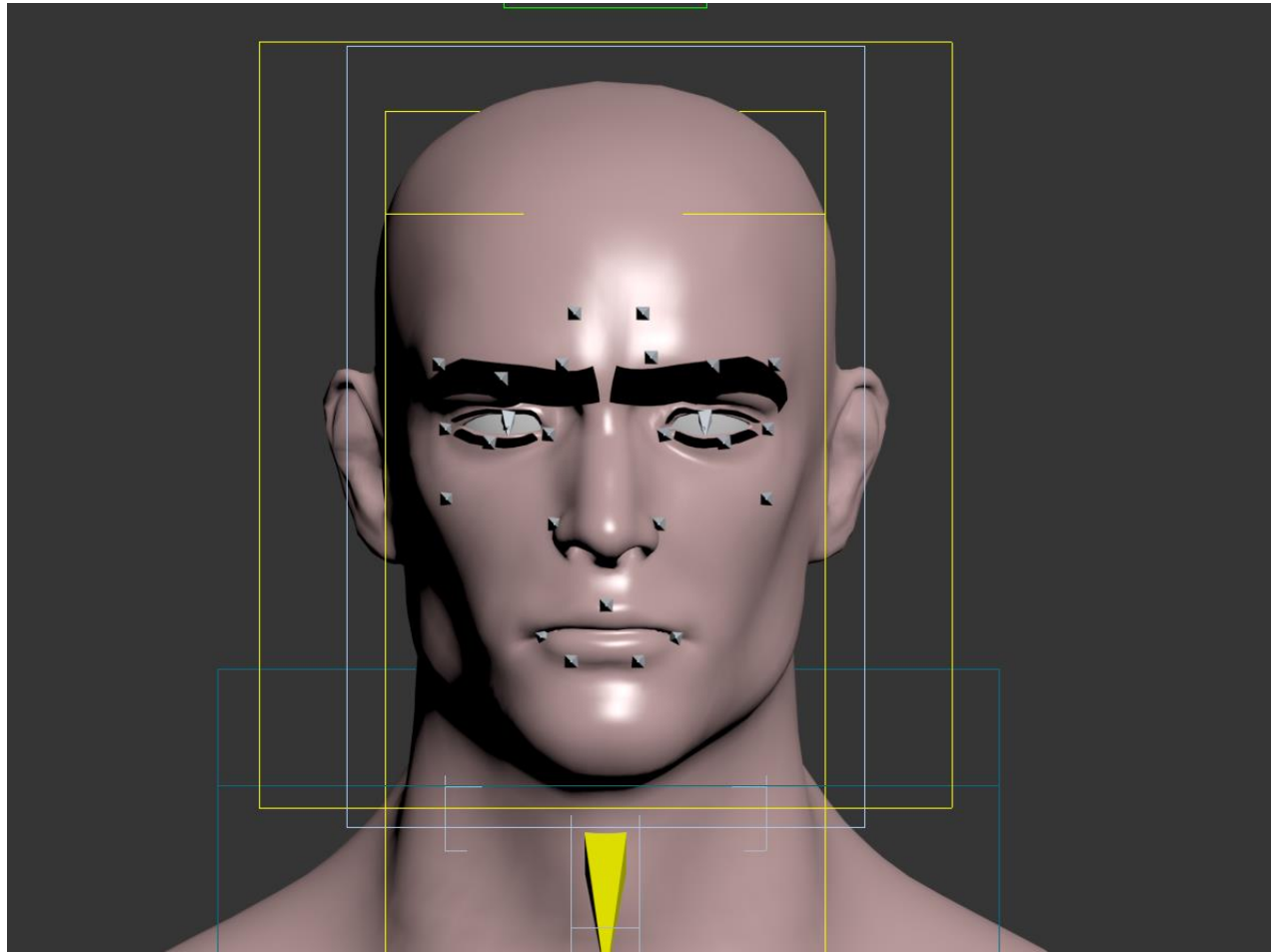
01. Rigging



목 갑옷의 경우에도 포지션을 담당하는 컨트롤러와
로테이션을 담당하는 컨트롤러를 분리하였고

마찬가지로 로테이션담당 컨트롤러의 트랜스폼을 참조하는 익스포트용 본을 제작하여 세팅
하는 것으로 최종 익스포트 되는 BONE의 갯수를 1개로 줄여 최적화에 성공했습니다.

01. Rigging



프로젝트 후반에 가서 시네마틱 영상속 플레이어가 잠깐 표정을 일그러트리는 장면이 필요했는데

이때 스켈레탈 메쉬를 하나 더생성하고 시네마틱용으로만 사용하여

본의 최적화를 신경쓰지 않아도 괜찮았기 때문에 다음과 같이 비합리적인 세팅의 페이스 리깅또한 진행
하였습니다.

01. Max Script

```
-- 선택된 객체들을 변수에 할당
Com = $Ref_SIMON_COM
Wb = $Weapon_bone
UR = $UpperRoot
Pelvis = '$Ref_Bip001 Pelvis'
R_Hand = '$Ref_Bip001 R Hand'

-- 부모-자식 관계 설정
Wb.parent = $Root
Com.parent = $Root
UR.parent = Pelvis

-- R_Hand의 자식으로 Wb 설정
Wb.parent = R_Hand

-- 파일 내보내기 작업 수행
actionMan.executeAction 0 "40011" -- File: Export File
```

개발과정에 필요했던 계층구조 자동 지정용 스크립트로

특정캐릭터의 애니메이션 파일에 일괄적으로 계층구조 변경을 해야하는이슈가 있었을때

해당 스크립트를 작성하여 대응하였습니다.

01. Max Script

```
(
    -- 기본 초기화 스크립트
    function resetAndConvertToPoly =
    (
        $.position = [0,0,0]
        resetXform $
        macros.run "Modifier Stack" "Convert_to_Poly"
    )

    global rot = eulerToQuat(eulerAngles 90 0 0)

    -- y_up 스크립트
    function Y_UP obj =
    (
        Select obj
        ResetXForm obj
        modPanel.addModToSelection (XForm ()) ui:on
        obj.modifiers[#XForm].gizmo.rotation += rot as quat
        rotate obj (inverse rot as quat)
        for obj in (selection as array) do (convertToPoly obj)
    )

    rollout lil "🐘의 익스포팅툴" width:180 height:140
    (
        groupBox grp1 "뭐할라고?" pos:[7,3] width:170 height:120
        button btnReset "영점에 돌래" pos:[25,23] width:127 height:34
        button btnExport "y축을 올려줘(유니티)" pos:[25,63] width:127 height:34

        on btnReset pressed do (
            -- "영점에 돌래?" 버튼 클릭 시 초기화 스크립트 실행
            resetAndConvertToPoly()
        )
    )
)
```

```
    -- 초기화 스크립트 실행 후 익스포트 창 띄우기
    local filePath = "C:\\\\경로\\\\파일이름.fbx" -- 내보낼 파일 경로 및 이름을 쓸
    local fileType = "FBX" -- 내보낼 파일 형식을 설정합니다

    -- 선택된 객체를 FBX 파일로 내보냅니다
    exportFile filePath fileType #noPrompt

    -- 내보내기가 완료된 후 메시지 표시
    messagebox "영점 이동 후 익스포트 완료" title: "🐘의 익스포팅툴"
)

on btnExport pressed do (
    selectedObjects = getCurrentSelection()
    if selectedObjects.count > 0 then (
        -- Y_UP 함수를 선택한 객체에 적용
        for obj in selectedObjects do Y_UP obj
    ) else messagebox "No object selected" title: "🐘의 익스포팅툴"
)

createDialog lil
)
```

코드는 다음과 같습니다.

감사합니다

