|  |  |  |  |
| --- | --- | --- | --- |
| MORPHOSIS 개발일지 23차 | | | |
| 기간 | 2019-10-08 ~ 2019-10-22 | 작성자 | 신재욱 |
| 작업 내용 | | | |
| 텍스처 배열의 사용법을 좀 확실하게 정리하고 가야겠다.  Root Parameter에서는 텍스처를 담기 위해 Descriptor Table을 사용한다. (Descriptor == 서술자 == View, 다 같은 의미이다)  Descriptor Table은 Descriptor Range 정보가 필요하다. 간단하게 말하면 ‘이러이러하게 쓸거다’라고 적어둔 구조체.  Descriptor Range 에는 이게 쉐이더 자원인지(SRV), 상수 버퍼 자원인지(CBV) 아니면 뭐 UAV 자원인지 RangeType을 정해줘야 하고,  Descriptor는 몇 개를 쓸건지(쉽게 말하면 텍스처 배열 할당할 때 몇 개로 할당할건지), 레지스터 첫 숫자는 뭘로 할건지(t4, t5 이런거) 등을 정할 수 있다. (ResisterSpace 등은 거의 쓸 일 없어서 설명 안 함)    요게 RootParameter. 맨 아래 요소는 어떤 쉐이더에서만 접근가능하게 할건지 정하는 값. (최적화 가능성이 있다지만 잘 모르겠다. 체감도 안 되고)    요게 DescriptorRange. 4번 레지스터 주소에 올라가는 버퍼의 값은 텍스처를 넣을 예정이므로 텍스처로 간주하고 해석해주세요~ 하는 의미이다. 근데 4번 레지스터 주소에 서술자 배열을 두고 그 배열의 길이는 5로 하겠다는 것이므로 그렇게 해석하면 된다.  이렇게 되면 t4 레지스터에는 Texture2D gtxtTexture[5]가 있을 예정이고, **실제로 사용해보니 gtxtTexture[0]으로 첫 번째 텍스처, gtxtTexture[1]로 두 번째 텍스처가 샘플러에 잘 들어오는걸 볼 수 있었다.** <- 이것이 궁금했던 부분.  근데 웃긴건 나는 텍스처들을 로드하고 그 텍스처들의 리소스뷰(==Descriptor)를 만들어주긴 했는데 실제 Render를 할 때는 첫 번째 텍스처만 SetGraphicsRootDescriptorTable() 해줬다는 점이다.  별다른 무언가 없이도 첫 텍스처의 다음 텍스처가 잡혀서 올라왔다는 건데 앞으로 텍스처 자원을 잘 쓸 수 있어야 그래픽 효과를 넣을 때 무리가 없으므로 이 부분을 확실히 하고 가기로 했다.  PS에서 Depth값을 얻어와서 그리는걸 해보자.      픽셀과 카메라의 거리를 구해서 반영.  그림자를 만들려면 일단 여러 가지 선행되야 하는 것이 있는데   * 텍스처에 렌더링을 할 수 있어야 함. * 텍스처 투영을 할 수 있어야 함. * 위를 바탕으로 섀도우 맵을 만들고 텍스처 투영을 할 수 있어야 함.   위의 세 가지이다.  텍스처에 렌더링 하는 것(Render To Texture, RTT)는 컴퓨트 쉐이더를 사용하여 텍스처 리소스에 렌더링을 하는 것을 의미한다. 컴퓨트 쉐이더를 사용하기 위해서는 먼저 기존의 PSO와 다른 Compute PSO를 사용해야 한다. 그것부터 만들어주자.    굿.  이제는 입력 텍스처 자원과 출력 텍스처 자원을 써야 한다. 입력은 기존 텍스처 사용방식과 동일하게 SRV로 생성해주면 된다. 출력은 조금 다른데 GPU에서 읽고 쓸 수 있어야 하기 때문에 읽기-쓰기를 지원한다는 의미의 ‘RW’가 붙는 자료형을 쓴다.  위의 자원들을 사용하려면 루트 파라미터를 수정해줘야 한다. | | | |