

심플 라이팅 샘플

*이 샘플은 Windows 10 기념일 업데이트 SDK (14393)*

# 설명

이 샘플은 정적 및 동적 Lambertian 조명을 사용하여 인덱싱 된 형상을 그리는 정적 Direct3D 11 버텍스, 인덱스 및 상수 버퍼를 만드는 방법을 보여줍니다.

이 샘플에서는 두 개의 조명 (흰색과 빨간색 중 하나)에 의해 큐브로 표현되는 대형 큐브를 렌더링합니다. 백색 라이트는 고정되어 있으며 빨간색 라이트는 중앙 큐브를 중심으로 회전합니다. 중앙 큐브도 회전합니다. 모션을 사용하면 각기 다른 각도에서 컬러 조명의 효과를 관찰 할 수 있습니다.



# 샘플 사용하기

|  |  |  |
| --- | --- | --- |
| 액션 | 게임 패드 | 키보드 |
| 종료 | 보기 버튼 | Esc |

# 구현 정보

이 샘플은 정점 셰이더 ("TriangleVS")와 두 개의 픽셀 셰이더 ("LambertPS", "SolidColorPS")를 사용하여 장면을 렌더링하는 세 개의 셰이더를 사용합니다. 컴파일된 셰이더 블롭은 CreateDeviceDependentResources에 로드된 다음 셰이더 리소스를 만들 때 참조됩니다. 모든 셰이더는 동일한 HLSL 포함 파일 "SimpleLighting.hlsli"에 정의되어 있으며, 세 개의 스텁 셰이더에는 이 파일이 포함되어 있습니다. 각 스텁 쉐이더는 세 개의 쉐이더 블롭을 생성하기 위해 다른 진입점에 대해 컴파일됩니다.

장면의 기하학의 정점의 6 개 쿼드를 나타내는 24 개의 정점에 대한 데이터로 채워지는 정적 버텍스 버퍼와 인덱스 버퍼로 구성됩니다. 이 2 개의 버퍼는 CreateDeviceDependentResources로 작성되어 D3D11\_SUBRESOURCE\_DATA 구조체의 초기화 데이터로 제공되는 데이터로 즉시 채워집니다. 이것은 DirectX 11의 초기화에 있어서 가장 효율적인 방법입니다.

이 매우 간단한 장면에서 모든 셰이더 상수는 다음을 포함하는 하나의 상수 버퍼로 함께 정리됩니다.

* 월드, 뷰 및 투영 행렬
* 가벼운 방향과 색상
* 단색

보다 복잡한 장면의 경우 상수가 얼마나 자주 업데이트 되는 지에 따라 상수를 여러 버퍼로 분할합니다.

대형 큐브와 빨간 라이트가 움직이기 때문에 어떤 상수는 매 프레임마다 업데이트 되어야 합니다. 상수 버퍼는 다음 전략을 사용하여 업데이트 됩니다.

1. D3D11\_MAP\_DISCARD를 사용해 ID3D11DeviceContext::Map을 불러와 D3D11\_MAPPED\_SUBRESOURCE를 생성합니다.
2. 모든 상수의 값을 매핑된 리소스에 복사합니다.
3. DrawIndexed를 호출하기 전에 리소스의 매핑을 해제합니다.

D3D11\_MAP\_DISCARD를 사용하는 경우

* 드라이버는 기본적으로 새로운 버퍼를 할당하므로 셰이더가 사용할 버퍼의 모든 상수를 업데이트해야 한다고 가정해야 합니다. 버퍼가 이전의 갱신으로부터의 유효한 데이터를 포함할 가능성이 있는 D3D11\_MAP\_WRITE\_NO\_OVERWRITE와 비교합니다.
* 이전에 매핑된 버퍼가 여전히 GPU에서 사용 중이면 새 버퍼가 생성됩니다. D3D11\_MAP\_DISCARD를 과도하게 사용하면 더 많은 메모리가 사용 가능해질 때까지 기다려야 하기 때문에 많은 시간이 들 수 있습니다.

## 시스템 요구사항

이 샘플은 Windows 8 이상과 호환됩니다.

이 샘플은 ([KB 2670838](https://support.microsoft.com/ko-kr/help/2670838/platform-update-for-windows-7-sp1-and-windows-server-2008-r2-sp1))이 설치된DirectX 11.1 런타임이 제공된 Windows 7 서비스 팩 1에서도 실행됩니다. 자세한 내용은 [이 블로그 게시물](https://blogs.msdn.microsoft.com/chuckw/2013/02/26/directx-11-1-and-windows-7-update/) 을 확인하십시오.

# 업데이트 기록

초기 출시 2018 년 4 월.