



언리얼 페스트 2024 서울

서브스트레이트

사실적인 재질 표현을 위한 차세대 머티리얼 시스템 알아보기

김대혁

디벨로퍼 릴레이션, 테크니컬 아티스트
에픽게임즈 코리아

박창현

디벨로퍼 릴레이션 리드
에픽게임즈 코리아

서브스트레이트

사실적인 재질 표현을 위한
차세대 머티리얼 시스템 알아보기

Section 1

서브스트레이트 소개

Section 2

서브스트레이트 사용법

Section 3

서브스트레이트 한계

Section 4

서브스트레이트 성능&디버깅

서브스트레이트 소개

서브스트레이트의 도입 이유

기존 머티리얼 시스템의 한계

더 확장된 표현을 위해

다른 툴들과의 호환성

기존 머티리얼 시스템의 한계

메탈릭의 한계

기존 머티리얼 시스템 = 메탈릭을 기반으로 한 물리기반렌더링
메탈릭은 물론 편리한 개념이지만
모든 물질을 메탈릭 & 논메탈릭으로 구별 불가능



기존 머티리얼 시스템의 한계

메탈릭의 한계

언리얼 엔진에서 스펙큘러는 비금속의 반사율

메탈릭이 0 일 때, 스펙큘러값이 1이어도 반사율은 8%
반사율은 F0이기 때문에, F0값은 0~0.08로 제한

F0은 IOR값을 통해 구할 수 있다
F0값이 0.08까지로 제한되어 있으면 IOR도 1.78로 제한

대다수의 재질은 IOR 1.78로 표현이 가능 하지만
IOR이 1.78을 넘어가는 케이스가 분명히 있다

스펙큘러

스펙큘러(Specular) 값은 0과 1 사이의 값을 받으며, 표면에 스펙큘러 라이드가 반사되는 정도를 제어합니다.

- 스펙큘러 값이 0이면 반사가 없습니다.
- 스펙큘러 값이 1이면 완전히 반사됩니다.

언리얼 엔진이 사용하는 기본 스펙큘러 값 0.08은 영광형의 스펙큘러 레프렉션에 따라입니다. 이 값은 대부분 내비리얼에서 상한하게 적용합니다.

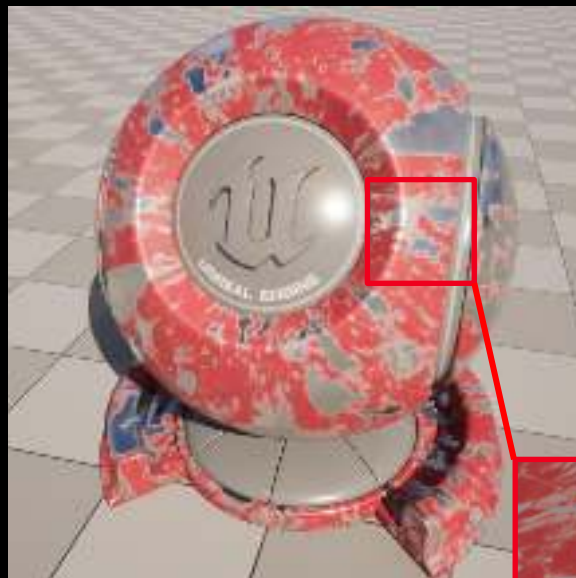
$$F(0^\circ) = \left(\frac{n - 1}{n + 1} \right)^2$$

Ruby	1,7700
Portland gem	1,7700
Extraordinarily flint	1,8800
Crystal	2,0000
Diamond	2,4170
Chromium oxide	2,7050
Copper Oxide	2,7050
Amorphous selenium	2,9200
Iodine crystals	3,3400

기존 머티리얼 시스템의 한계

픽셀당 하나의 셰이딩 모델만
표현 가능

메탈릭이 0에서 1로 변화할 때 생기는 아티팩트



더 확장된 표현

복잡한 재질들의 표현

?? : 코팅된 탄소섬유를 표현해주세요. 탄소섬유와 코팅된 외장의 스펙쿨러가 모두 보이게 해주세요

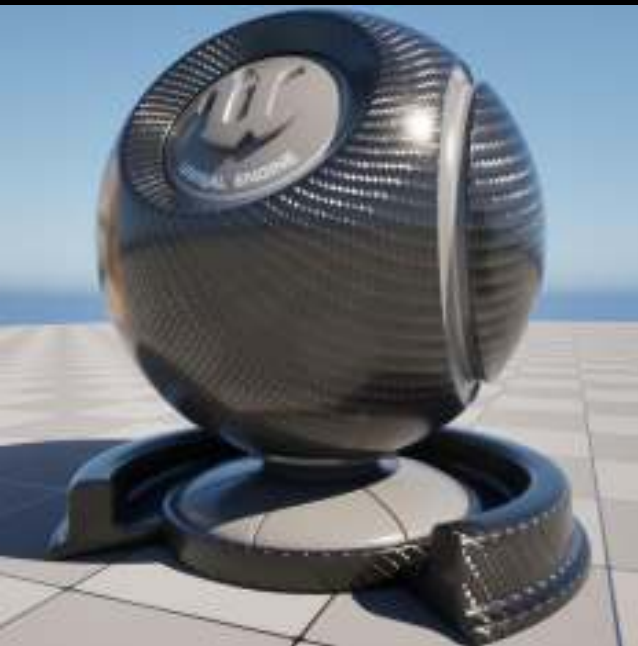
?? : 데칼은 쓰지말고, 욕실바닥 위에 피가 묻은 걸 표현해주세요. 피를 통해서 원래 욕실바닥이 살짝 보이게 해주세요

?? : 얼음속에 돌을 표현해주세요, 근데 얼음 속 돌 깊이에 맞춰서 색상이 어두워지고, 깊이가 깊을수록 불투명하게 보이게 해주세요



더 확장된 표현

서브스트레이트는 가능합니다



더 확장된 표현

기존 셰이딩 모델을 통해서라면..

고도로 커스텀된 새로운 셰이딩 모델이 필요

새로운 셰이딩 모델의 무분별한 추가

상황이 지속되면 퍼포먼스 & 유지보수의 문제발생!

다른 툴들과의 호환성

많은 표준 머티리얼들

Autodesk Standard Surface
Adobe Standard Material
Nvidia MDL
Material X

레이어링 개념을 가지고 있어서 좋지만
그대로 쓰기에는 무겁고 제한이 많다

표준 머티리얼들 호환 + 우리의 입맛에 맞게 = 서브스트레이트

Autodesk/
standard-surface



Adobe Standard Material



MATERIAL X

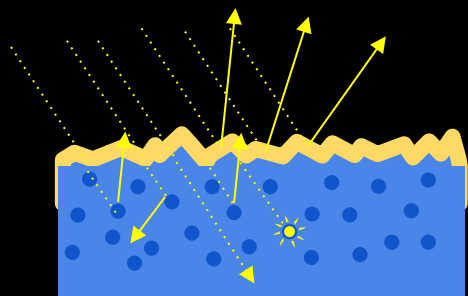
그래서 결국
서브스트레이트란?

창의성에 제한없이

작업효율을 높일 수 있는

멀티 BSDF 프레임워크

서브스트레이트의 세가지 주요 개념



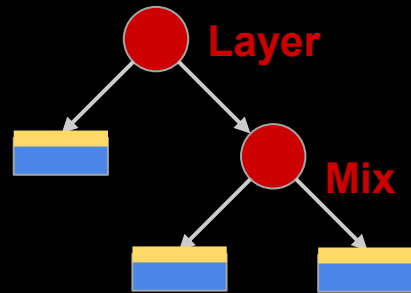
슬랩(Slab)

+



연산자(Operator)

=



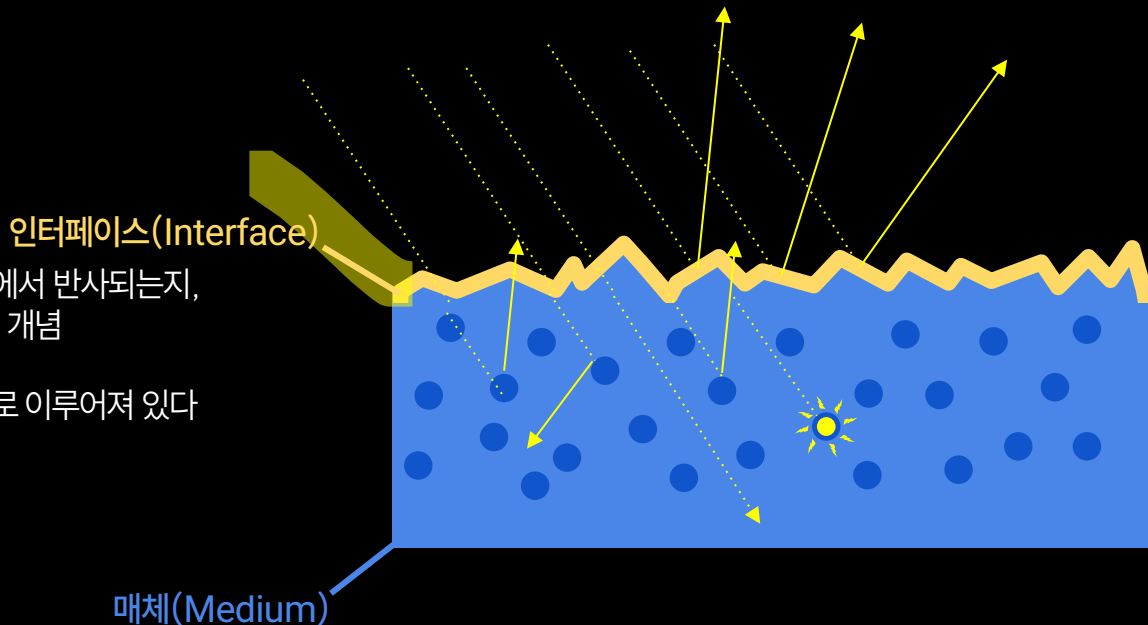
트리(Tree)

슬랩(Slab)

물리적인 단위를 쓰는 원칙적인 BSDF

여러가지 파라미터를 통해서 빛이 인터페이스에서 반사되는지,
매체에 흡수되는지, 통과하는지 등을 결정하는 개념

인터페이스(Interface)와 매체(Medium)로 이루어져 있다



슬랩(Slab)

물리적인 단위를 쓰는 원칙적인 BSDF

서브스트레이트의 슬랩 노드
여러가지 인풋을 인터페이스와 매체로 구별 할 수 있다

매체(Medium)

인터페이스(Interface)

매체(Medium)

인터페이스(Interface)



슬랩(Slab) 인터페이스(Interface)

F0 & F90

F0 표면에 빛이 수직으로 입사할 때 반사되는 빛의 비율
F90 표면에 빛이 수평으로 입사할 때 반사되는 빛의 비율

보통 금속은 F0이 1에 가깝고 비금속은 F0이 0.08이하로 낮다
이걸 기준으로 메탈릭이냐 아니냐로 나누었던 것

F90은 거의 모든 재질이 1에 가깝다



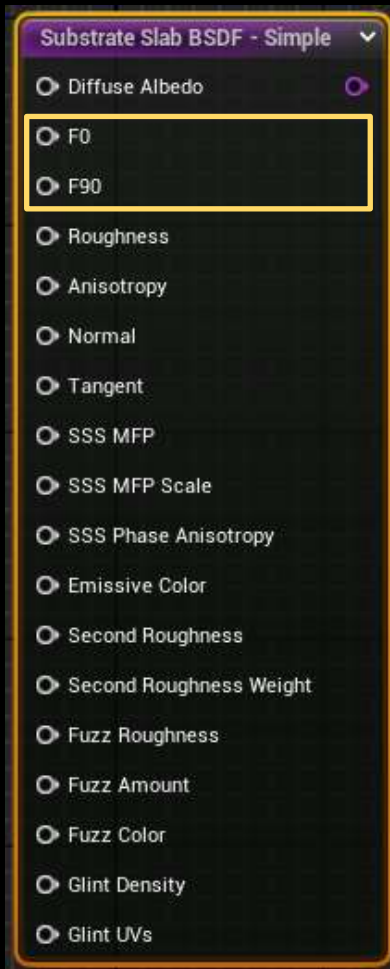
The PBR Guide by Allegorithmic - Part 1



F0(1, 1, 1) F90(1, 1, 1)



F0(0, 1, 1) F90(1, 0, 0)



슬랩(Slab) 인터페이스(Interface)

러프니스(Roughness) 애니소트로피(Anisotropy)

기존 레거시 방식하고 크게 바뀐 것이 없다

러프니스는 표면의 거칠기

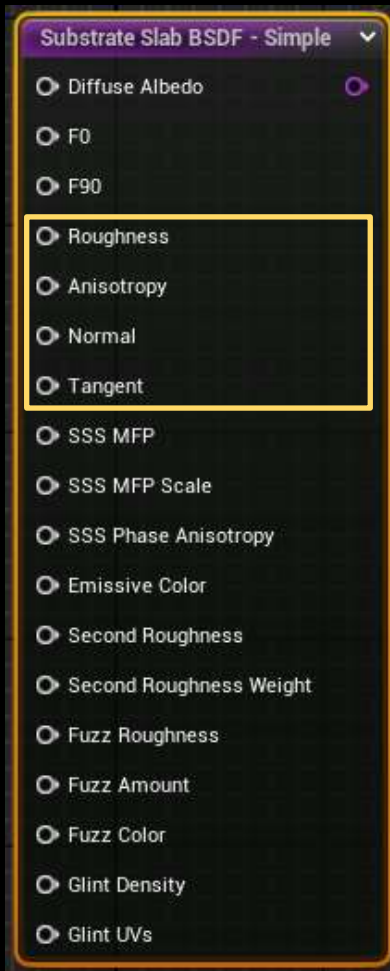
애니소트로피는 -1~1의 값을 가지며
양수일때는 탄젠트 벡터를 따라서 스펙쿨러가 늘어나고
음수일때는 탄젠트 벡터의 수직방향을 따라서 스펙쿨러가 늘어난다



러프니스



애니소트로피

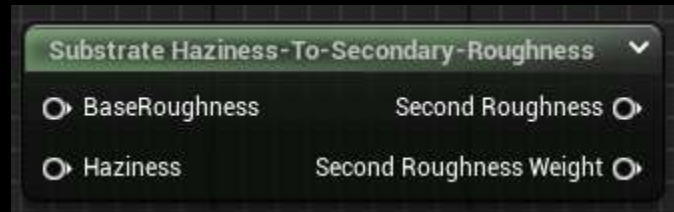


슬랩(Slab) 인터페이스(Interface)

세컨더리 러프니스(Second Roughness)

기본적으로 스펙큘러는 GGX를 따르지만,
좀 더 꼬리가 긴 스펙큘러를 표현하고자 할 때 사용한다

헬퍼 노드 사용을 권장



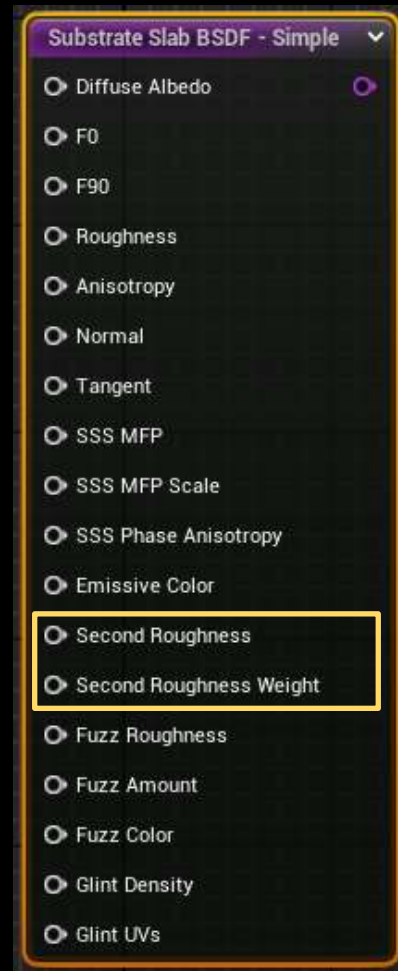
*Second Roughness Weight가 1이면
기존 러프니스는 디퓨즈의 러프니스만 담당



러프니스만 있을 때



세컨더리 러프니스 추가



슬랩(Slab) 인터페이스(Interface)

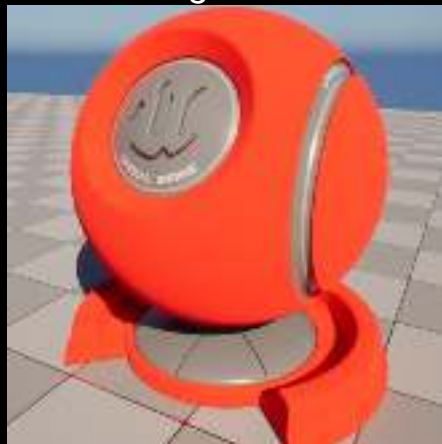
디퓨즈 러프니스?

Second Roughness Weight가 1일 때
기존 러프니스는 디퓨즈에만 영향을 끼침

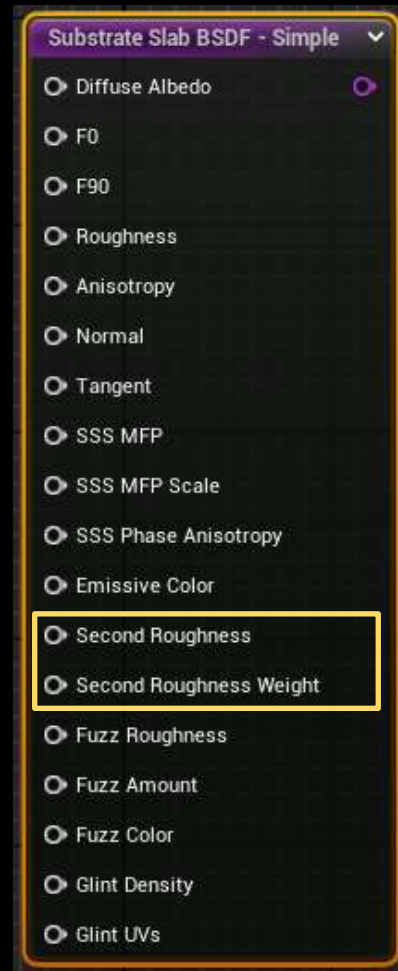
금속 재질엔 거의 영향이 없다



Roughness 0



Roughness 1



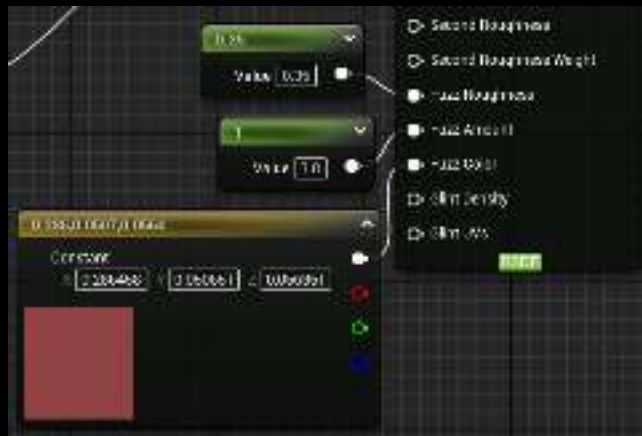
슬랩(Slab) 인터페이스(Interface)

퍼즈(Fuzz)

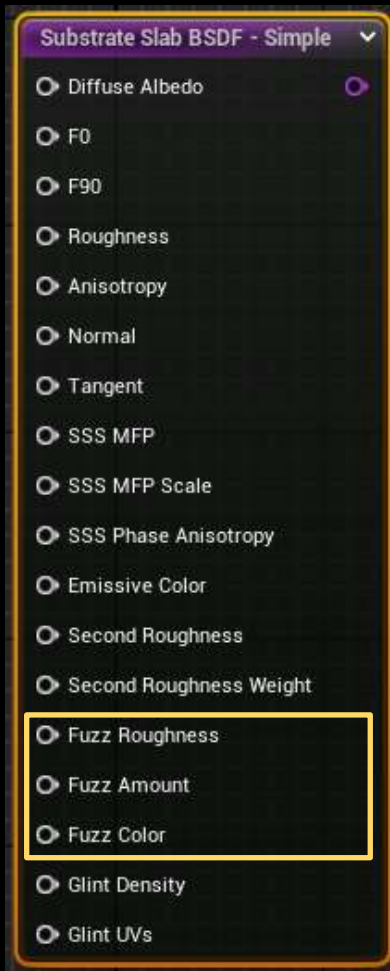
표면이 매우 미세하고 부드러운 재질들 표현에 용이
예) 벨벳, 카펫, 스웨터, 이끼 등



Velvet



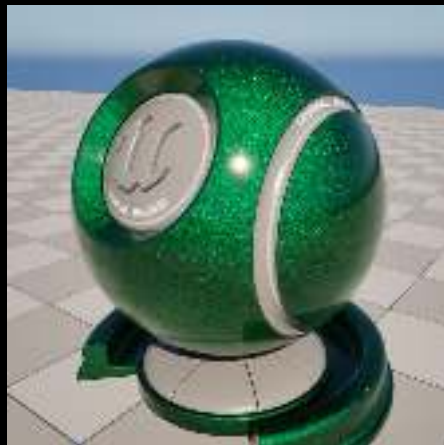
Node



슬랩(Slab) 인터페이스(Interface)

글린트(Glint)

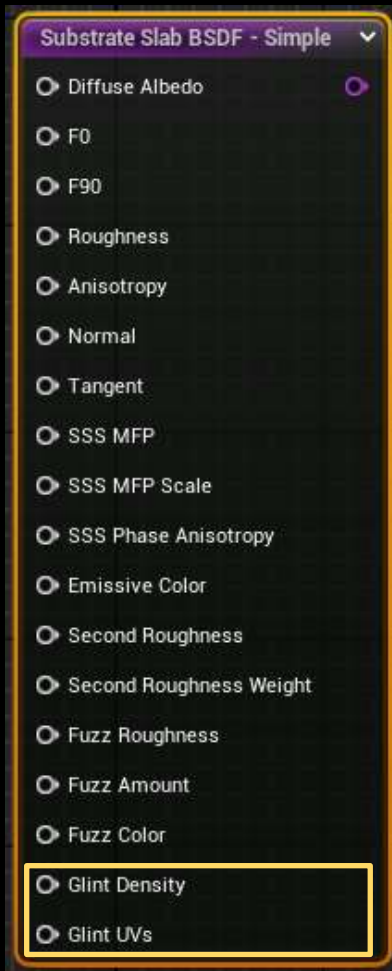
자동차 페인트, 눈같이 반짝이는 효과에 사용한다
거리에 따라서 자연스럽게 Fade 됨



Car Paint



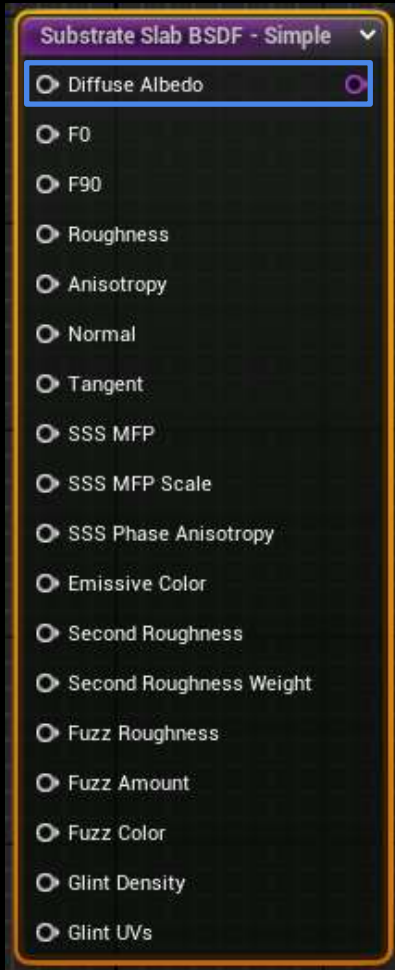
Snow



슬랩(Slab) 매체(Medium)

디퓨즈 알베도(Diffuse Albedo)

얼핏 생각해서 디퓨즈 알베도는 인터페이스(Interface)만의 특성일 것 같지만 매체(Medium)와도 관련이 있다
알베도는 다중산란이후 전체적인 매질의 반사율이기 때문



슬랩(Slab) 매체(Medium)

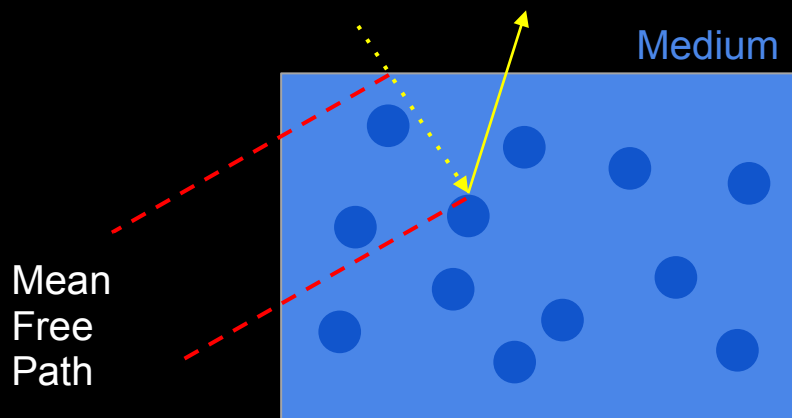
MFP(Mean Free Path)

광자가 매체(Medium)의 입자에 도달하기 전까지의 자유 이동 거리
피부나 식생 표현에 적합하다

*서브스트레이트에서 단위는 cm

	albedo			dirt [mm]			ior
apple	0.846	0.841	0.528	6.96	6.40	1.90	1.3
chicken1	0.314	0.156	0.126	11.61	3.88	1.75	1.3
chicken2	0.321	0.160	0.108	9.44	3.36	1.79	1.3
cream	0.976	0.900	0.725	15.08	4.66	2.54	1.3
ketchup	0.164	0.006	0.002	4.76	0.58	0.39	1.3
warble	0.830	0.791	0.753	8.51	5.57	3.95	1.5
potato	0.764	0.613	0.213	14.27	7.23	2.04	1.3
skimilk	0.815	0.813	0.682	18.42	10.44	3.50	1.3
skin1	0.436	0.227	0.131	3.67	1.37	0.68	1.3
skin2	0.623	0.433	0.343	4.82	1.69	1.09	1.3
spectralon	1.000	1.000	1.000	inf	inf	inf	1.3
wholesilk	0.908	0.881	0.759	10.90	6.58	2.51	1.3

MFP Table(Pixar Renderman)



Substrate Slab BSDF - Simple

- Diffuse Albedo
- F0
- F90
- Roughness
- Anisotropy
- Normal
- Tangent
- SSS MFP
- SSS MFP Scale
- SSS Phase Anisotropy
- Emissive Color
- Second Roughness
- Second Roughness Weight
- Fuzz Roughness
- Fuzz Amount
- Fuzz Color
- Glint Density
- Glint UVs

슬랩(Slab) 매체(Medium)



With MFP

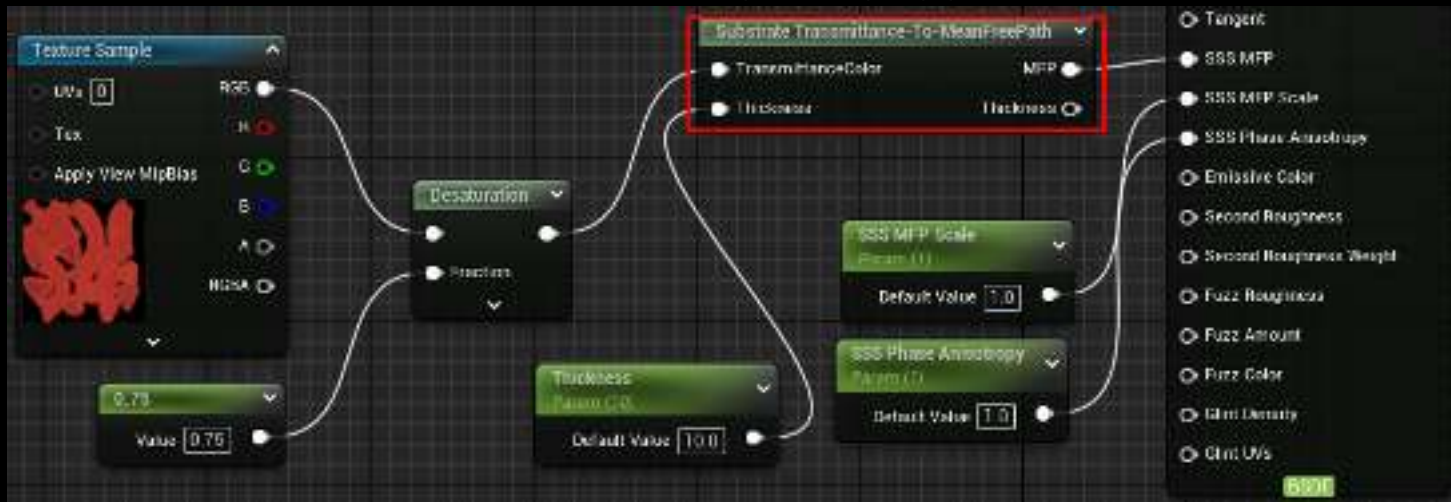


Without MFP

슬랩(Slab) 매체(Medium)

MFP(Mean Free Path)

MFP는 색상이 아니고 빛의 이동에 대한 측정이기 때문에 MFP에 색상을 바로 넣는 것은 추천하지 않음
Substrate Transmittance-To-MeanFreePath 헬퍼 노드를 사용하자



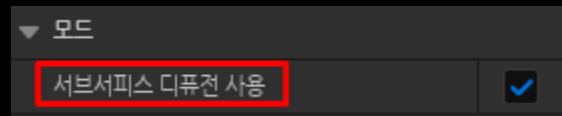
슬랩(Slab) 매체(Medium)

MFP(Mean Free Path)

서브서피스 디퓨전 체크가 되어있으면
Light Diffusion(SSS Diffusion)을 실행

체크가 되어있지 않으면
Wrap-Approximation을 실행

체크가 되어있지 않으면 기존의 레거시와 동일한 셰이딩
식물정도의 얇은 표면은 체크 해제를 추천



서브서피스 디퓨전 On



서브서피스 디퓨전 Off

슬랩(Slab) 매체(Medium)

MFP(Mean Free Path)

투명한 재질의 표현에도 MFP를 사용한다
색상이 있는 유리를 표현하고 싶으면
TranslucentColoredTransmittance로



MFP 0.01



MFP 1

슬랩(Slab) 매체(Medium)

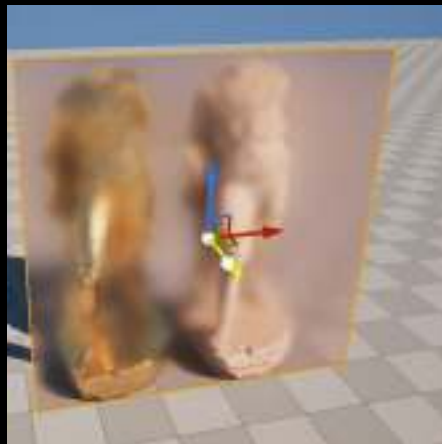
MFP(Mean Free Path)

투명한 재질의 러프니스는 뒤에 보이는 물체를 흐리게 만들 수 있다

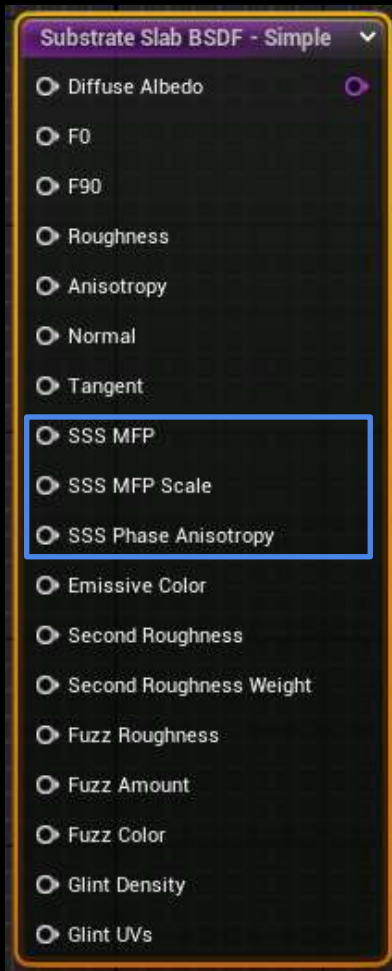
러프니스가 있는 투명한 재질은 거리가 멀어질수록 뒤에 보이는 물체를 흐리게 만들 수 있다



러프 리프렉션



거리를 통한 러프 리프렉션



연산자(Operator)

여러 슬랩을 어떻게 조합할지 결정하는 연산자

Vertical Layer, Horizontal Blend, Coverage Weight

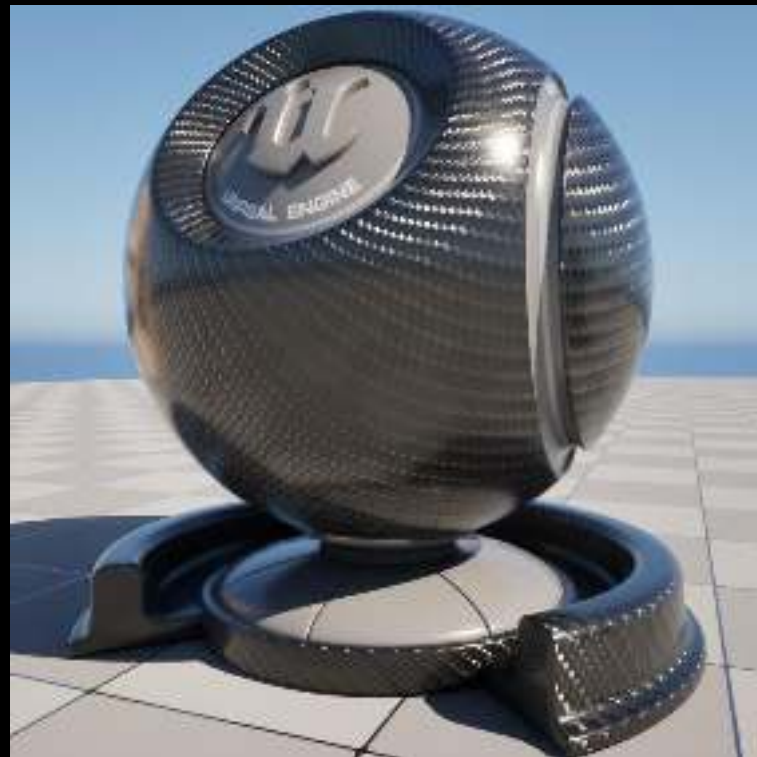
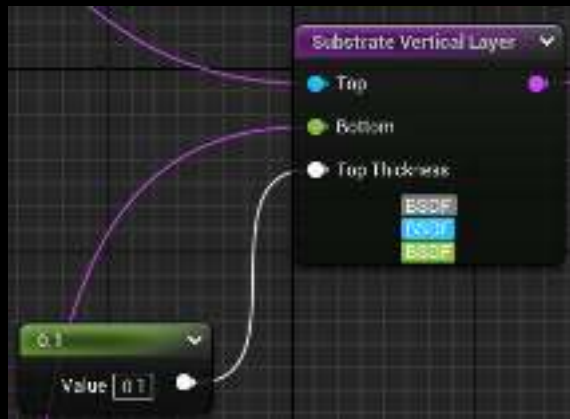


연산자(Operator)

Vertical Layer

수직으로 슬랩을 쌓는다
두 개 이상의 BSDF를 동시에 보여주고 싶을 때
ex) 코팅된 탄소섬유, 자동차 표면

*Thickness의 단위는 cm



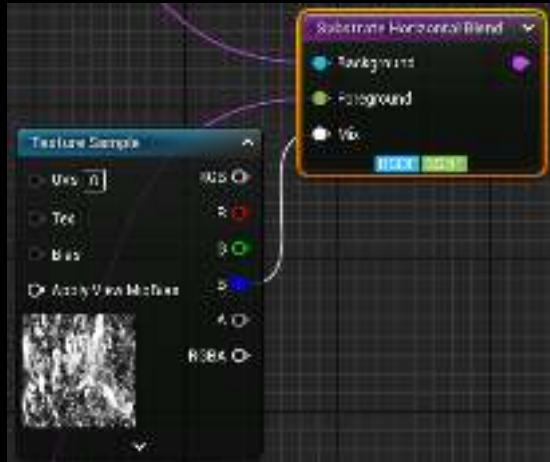
Coated Carbon Fiber



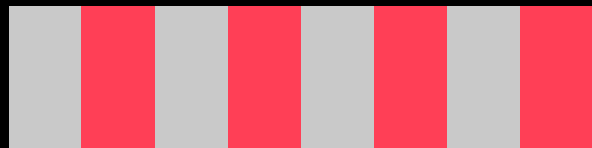
연산자(Operator)

Horizontal Blend

수평으로 슬랩을 배치한다
슬랩들을 Lerp한다고 생각
한 표면에 하나의 BSDF만 나올 때
ex) 벗겨진 페인트



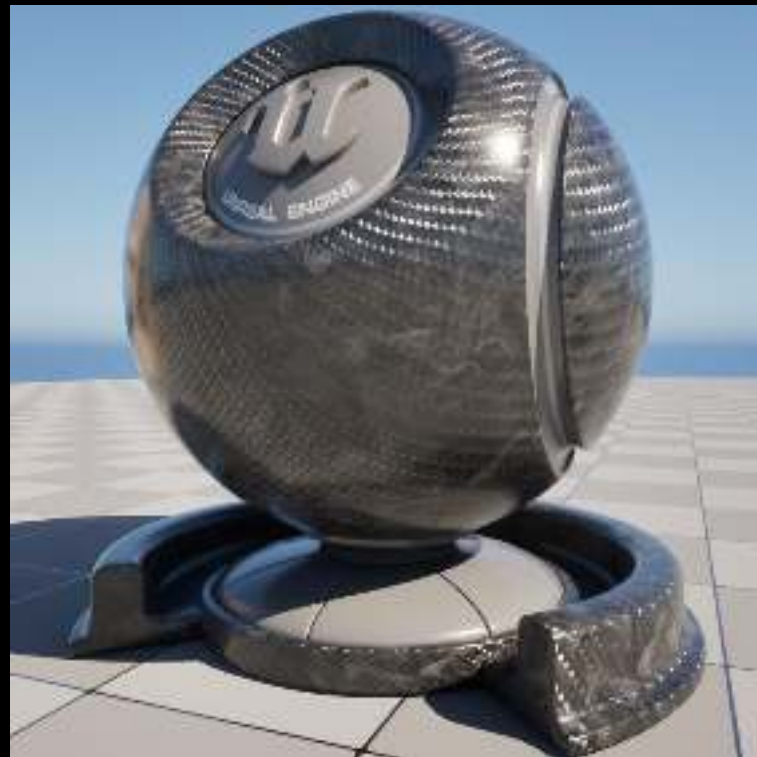
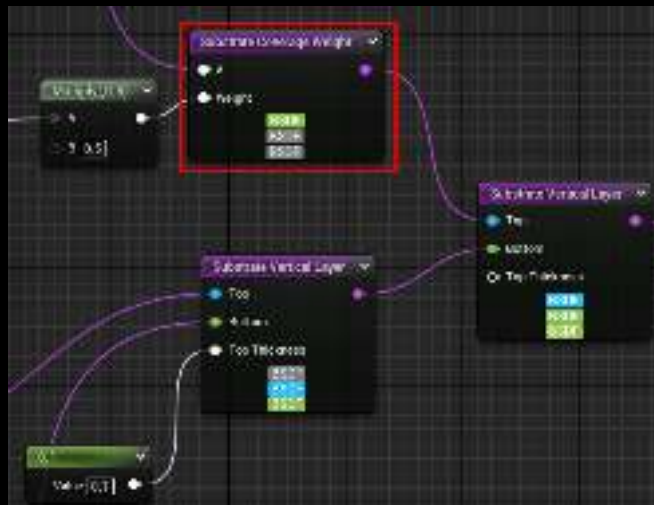
Metal with Paint



연산자(Operator)

Coverage Weight

마스킹을 통해서 일부만 보여주고 싶을 때,
주로 슬랩 구성 후에 곱해준다
Ex)핏자국, 먼지 등등



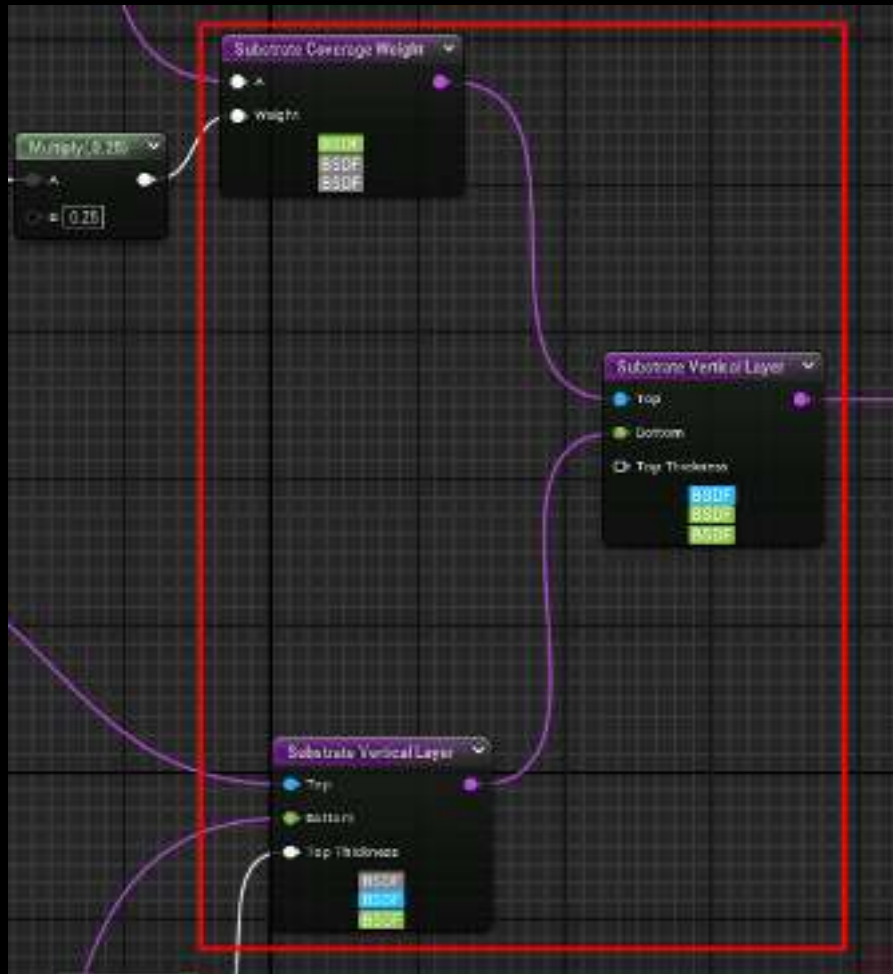
Coated Carbon Fiber with Dust



트리(Tree)

최종적으로 슬랩이 어떻게
구성되어 있는지

연산할 때 참조된다. 더 자세한 내용은 뒷부분에서



서브스트레이트 사용법

서브스트레이트 활성화

프로젝트 셋팅

엔진 - 렌더링 에서 서브스트레이트 항목 모두 활성화

▼ 엔진 - 렌더링

렌더링 설정들입니다.

▼ 서브스트레이트

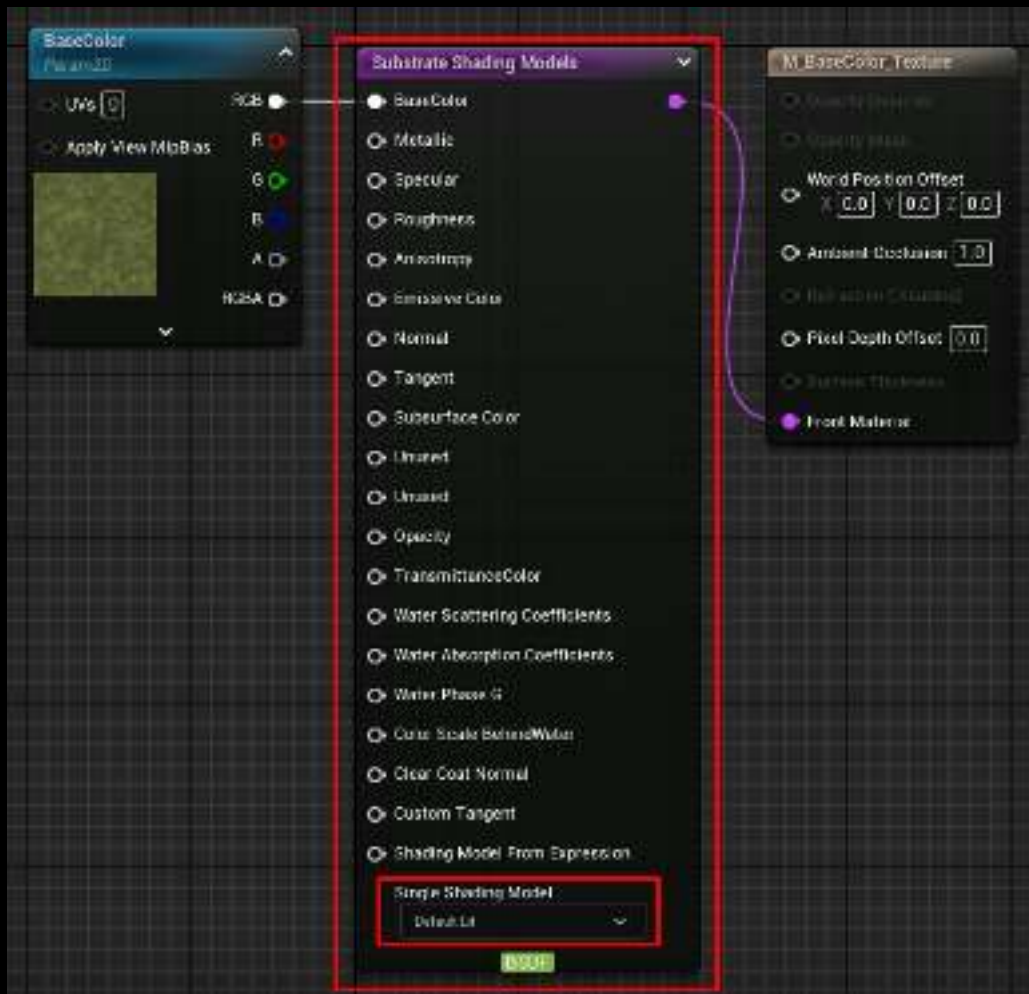
서브스트레이트 머티리얼(실험단계)	<input checked="" type="checkbox"/>
서브스트레이트 불투명 머티리얼 러프 리프렉션	<input checked="" type="checkbox"/>
서브스트레이트 고급 시각화 셰이더	<input checked="" type="checkbox"/>

서브스트레이트 활성화

기존 재질들

서브스트레이트 항목들을 체크 하고 엔진을 재부팅 하면 자동으로 변환 되어서 보존된다

하지만 실험 단계이니 프로젝트 백업 추천



서브스트레이트의 특징 (아티스트 입장에서)

메탈릭이 없다

가장 중요한 내용, 실제 수치를 넣어줘야 함

기존에 어려웠던 표현들 하기가 용이하다

Iridescent, 클리어 코트, 불투명 유리, 섬유 등

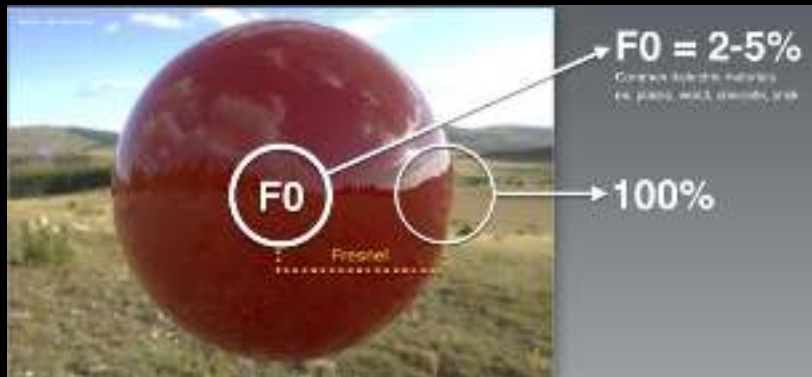
새로운 용어들을 알아야 한다

F0, F90, MFP, 세컨더리 러프니스, 퍼즈(Fuzz), 글린트(Glint) 등

메탈릭이 없다?

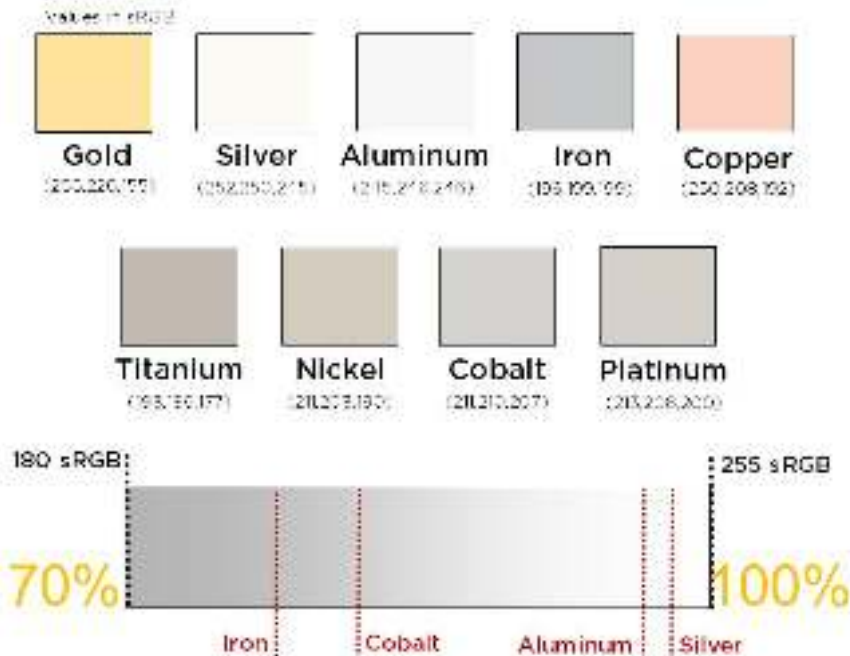
금속표현을 어떻게 하죠

F0을 통해서 금속을 표현할 수 있다
F0에 들어가는 값이 금속의 색상



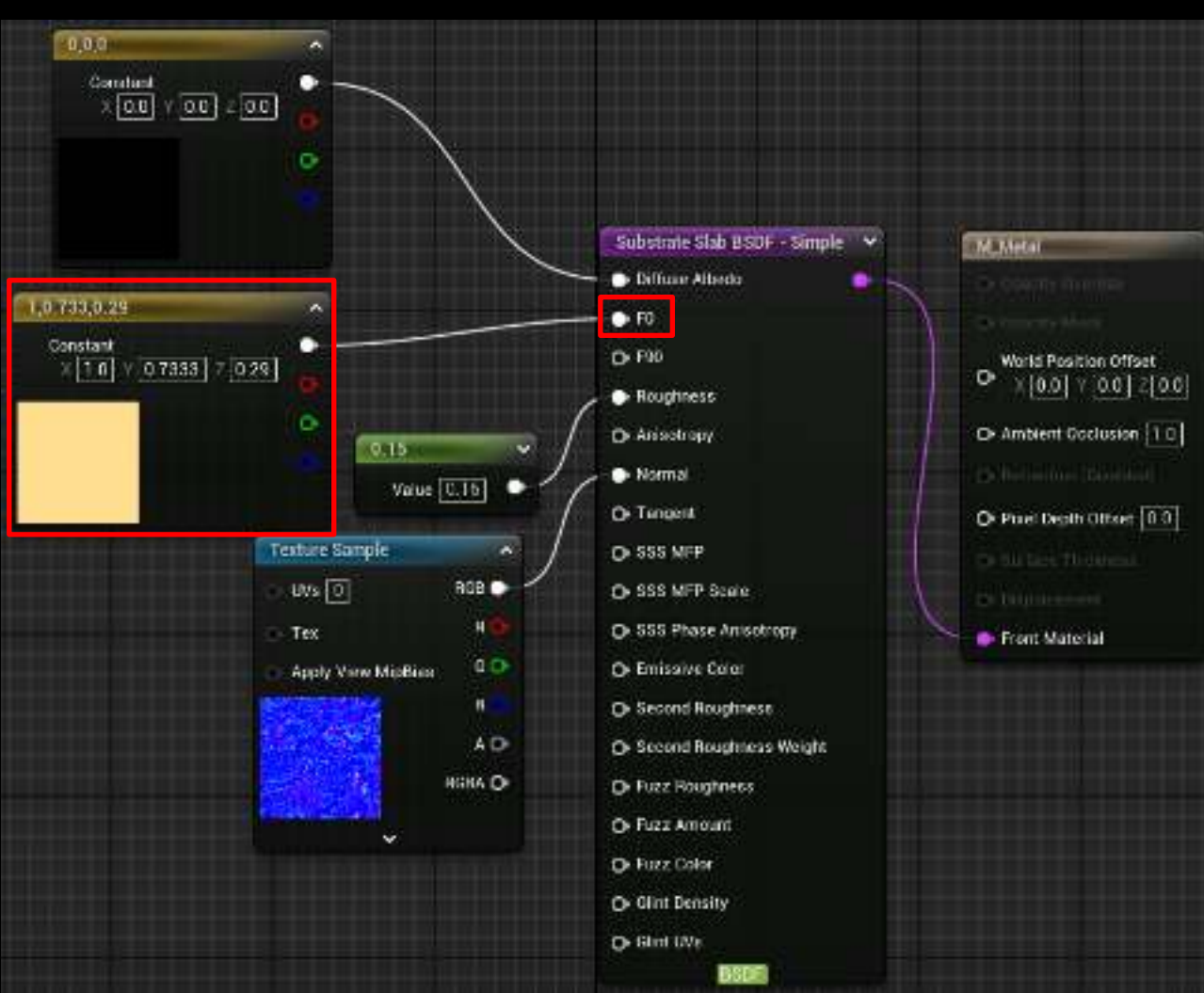
The PBR Guide by Algorithmic - Part 1

Metal



Conversions from sRGB to linear space are using the gamma 2.2
I support the sRGB (2000) aperture and also use the 1.45

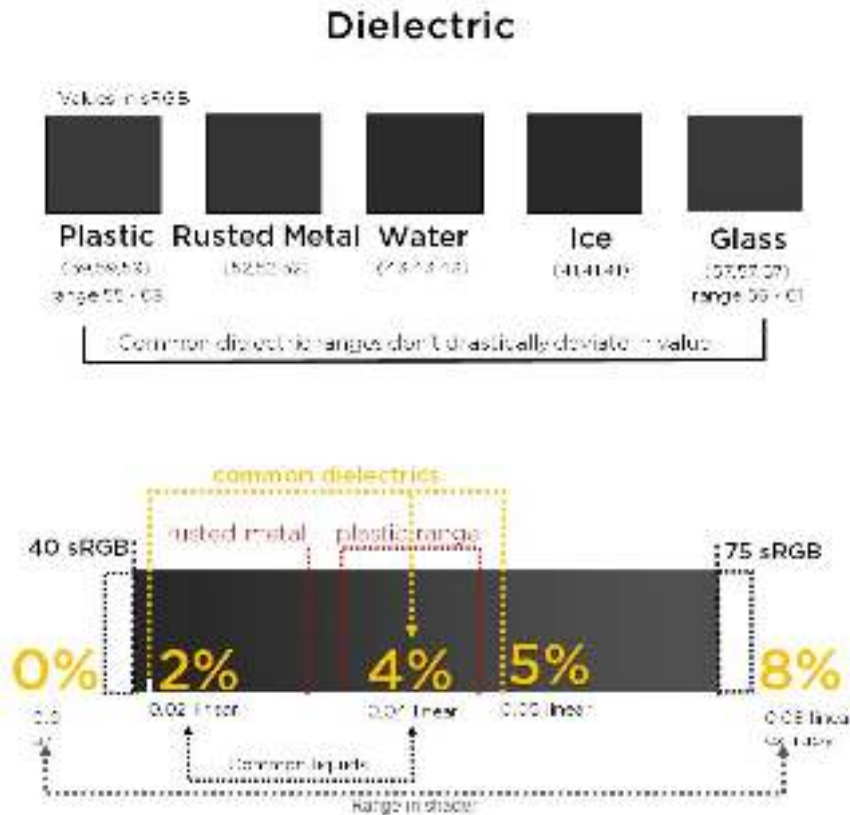
The PBR Guide by Algorithmic - Part 2



논 메탈릭은?

기존과 비슷합니다

디퓨즈 알베도에 베이스컬러 텍스처 입력
F0으로 반사율 조절(0~0.08)



Diffuse Albedo
Param2D

UVs [0]

Apply View MipBias



RGB

R

G

B

A

RGBA

Normal
Param2D

UVs [0]

Apply View MipBias



RGB

R

G

B

A

RGBA

Mask
Param2D

UVs [0]

Apply View MipBias



RGB

R

G

B

A

RGBA

Substrate Slab BSDF - Simple

- Diffuse Albedo
- F0
- F90
- Roughness
- Anisotropy
- Normal
- Tangent
- SSS MFP
- SSS MFP Scale
- SSS Phase Anisotropy
- Emissive Color
- Second Roughness
- Second Roughness Weight
- Fuzz Roughness
- Fuzz Amount
- Fuzz Color
- Glint Density
- Glint UVs

BSDF

AD
Param0D

Default Value [0.0]

M_nonMetallic

Specular Transmittance

Specular Mask

World Position Offset

X [0.0] Y [0.0] Z [0.0]

Ambient Occlusion

Refraction Disabled

Pixel Depth Offset [0.0]

Surface Thickness

Displacement

Final Material

Lerp(1,_)

A [1.0]

B

Alpha

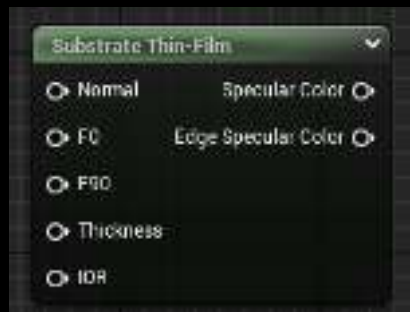


Iridescent

비눗방울 효과

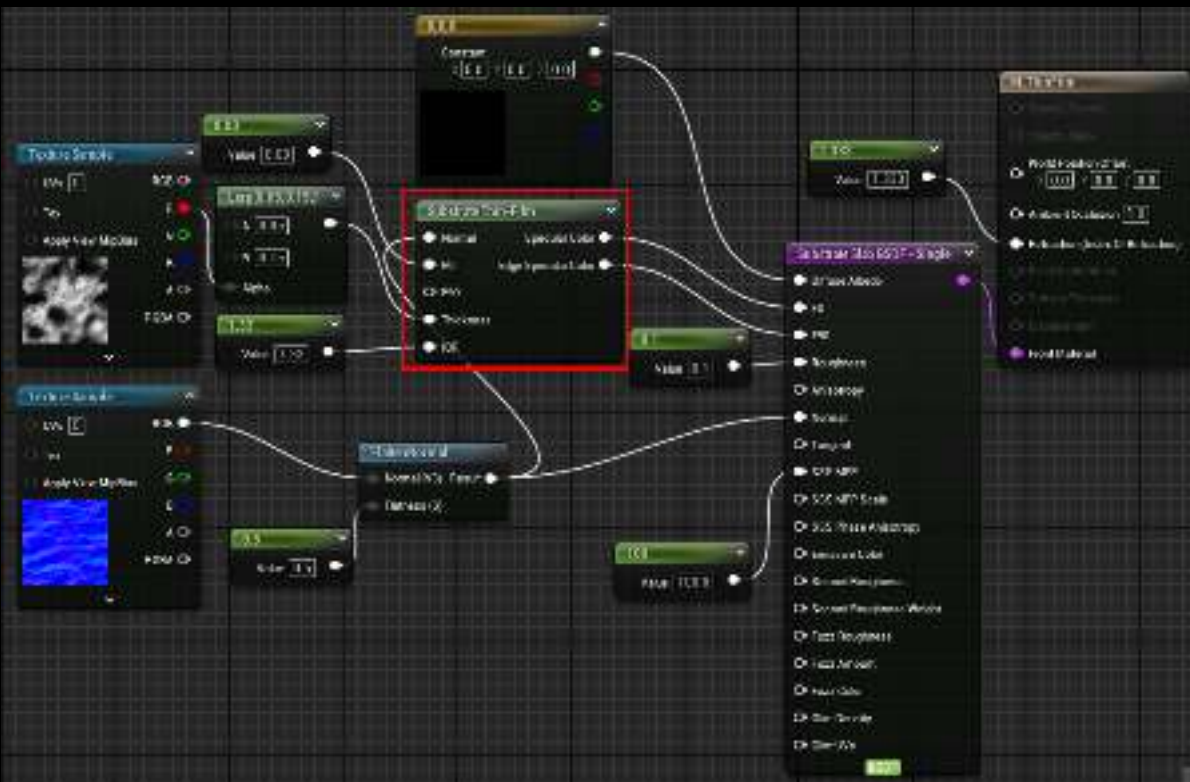
표면의 미세구조에서 파동의 간섭이 일어나서 생기는 광학적인 효과

기존 셰이딩 모델에서는 여러 수식들을 통해서 힘들게 구현하였다면 서브스프레이트에선 쉽게 가능



*Thickness가 1일 때 10마이크로 미터(0.01mm)

A screenshot of a Wikipedia article titled 'Iridescence'. The page has a light blue header with the title 'Iridescence' and a language selector '40 languages'. Below the header, there are tabs for 'Article' and 'Talk', and a row of links: 'Read', 'Edit', 'View history', and 'Tools'. The main text starts with 'From Wikipedia, the free encyclopedia' and a note: '*Iridescent* redirects here. For the Linkin Park song, see Iridescent (song). For the Brockhampton album, see Iridescence (album)'. The main body of text defines 'Iridescence' (also known as 'goniochromism') as the phenomenon of certain surfaces that appear to change color as the angle of view or the angle of illumination changes. It explains that iridescence is caused by wave interference of light in microstructures or thin films. Examples include soap bubbles, feathers, butterfly wings, and seashell nacre. It also mentions 'Pearlescence' as a related effect where some or most of the reflected light is white. To the right of the text is an image of a soap bubble showing iridescence, with a caption 'Iridescence in soap bubbles' and a small icon in the bottom right corner of the image area.



Blend Mode

TranslucentGreyTransmittance ▾

Lighting Mode

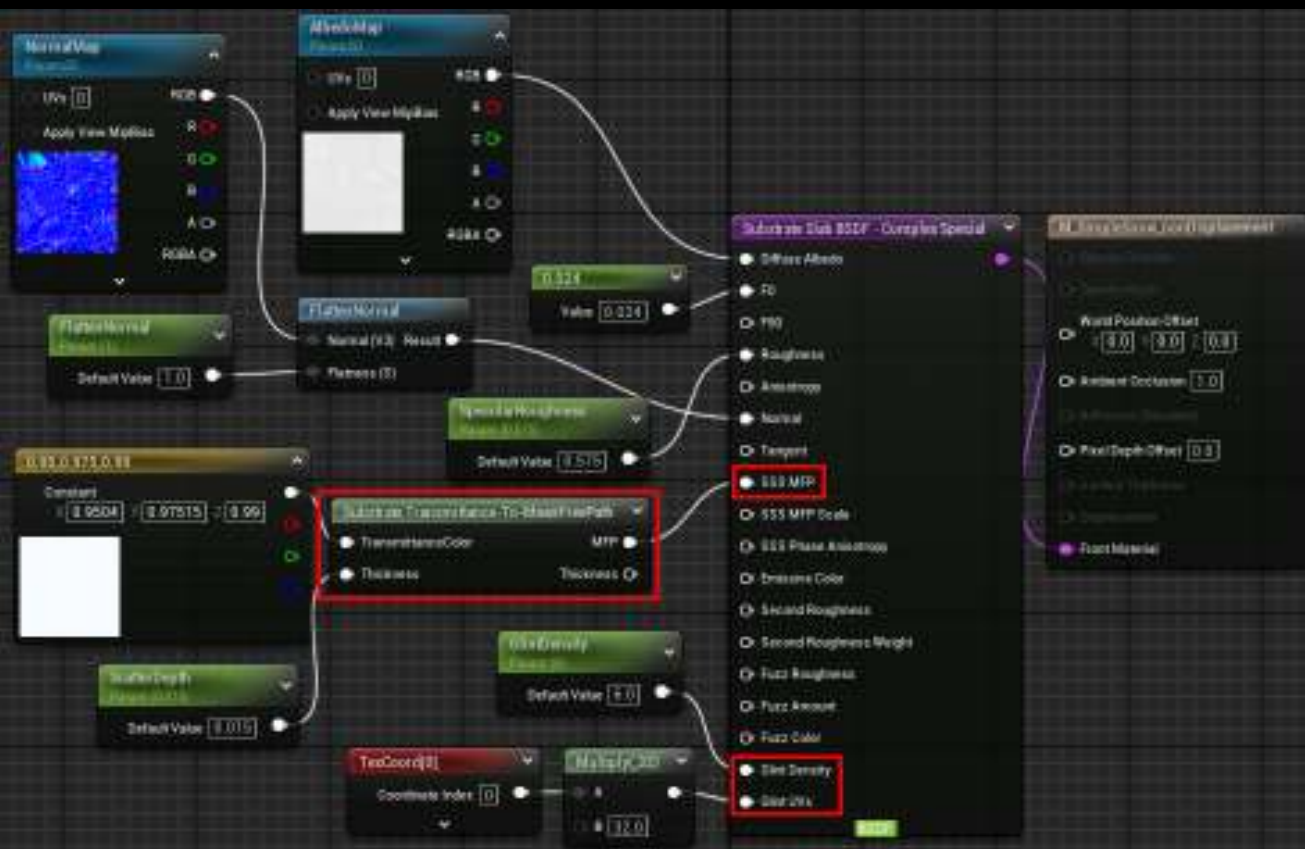
Surface ForwardShading ▾

눈덩이

서브서피스 스퀘터링 + 글린트

눈은 서브스피스 스퀘터링과 반짝이는 미세얼음을 같이 표현해야 해서 어렵지만 서브스트레이트에선 SSS MFP와 글린트(Glints)를 조합하여 어렵지 않게 가능하다





벨벳

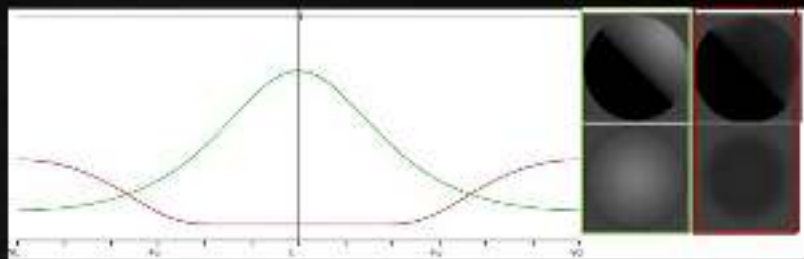
뒤집힌 스펙큘러 로브

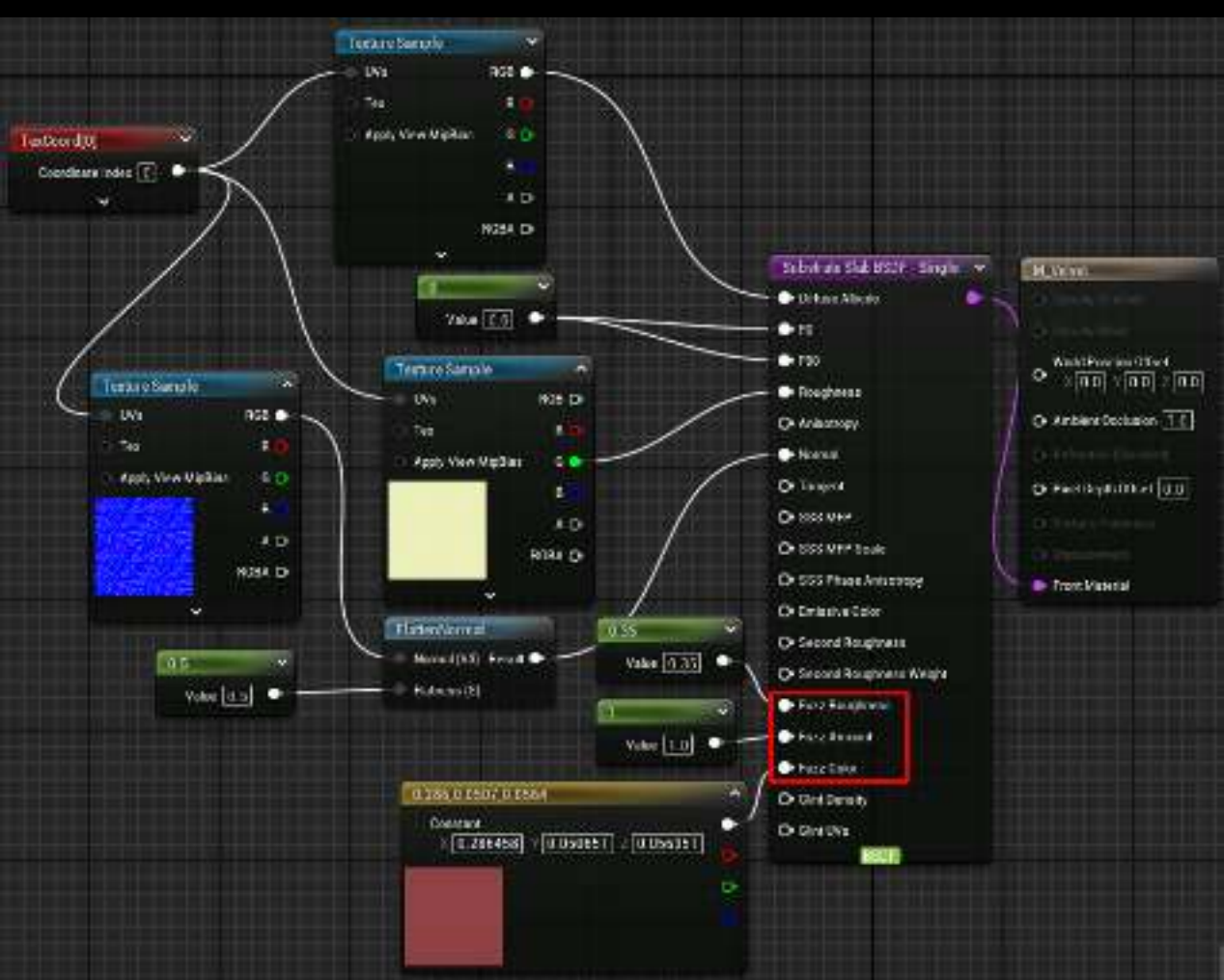
옷감은 스펙큘러 그래프가 일반적인 물질과 다르게 뒤집혀져 있어서 새로운 셰이딩 모델이 꼭 필요하다

서브스트레이트이션 퍼즈(fuzz)속성을 이용해서 어렵지 않게 할 수 있다

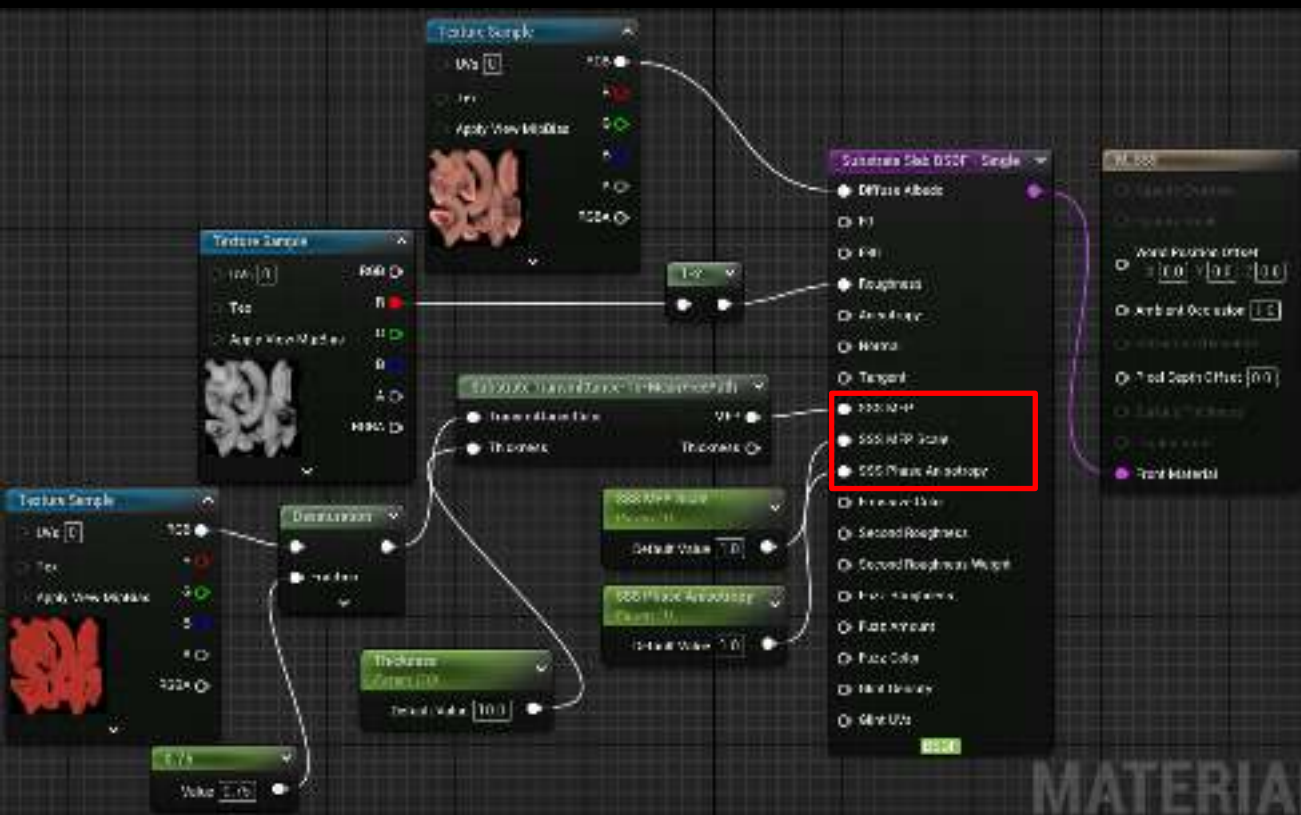
Cloth Shading

- Inverted Gaussian for asperity scattering
- Translation from origin to give more specular at front facing angles
- No geometry term

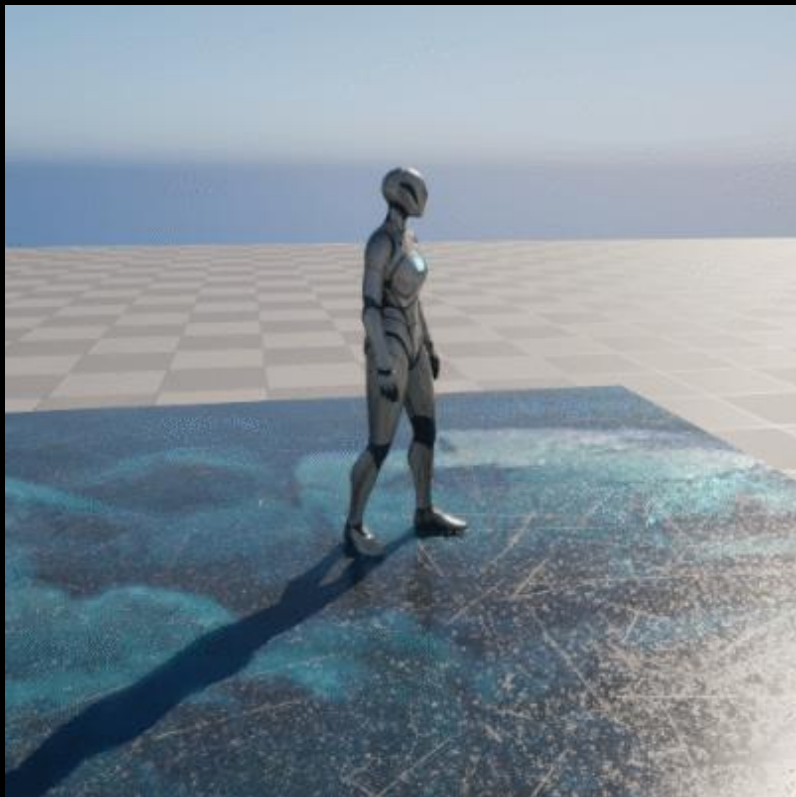




피부



얼음속 돌

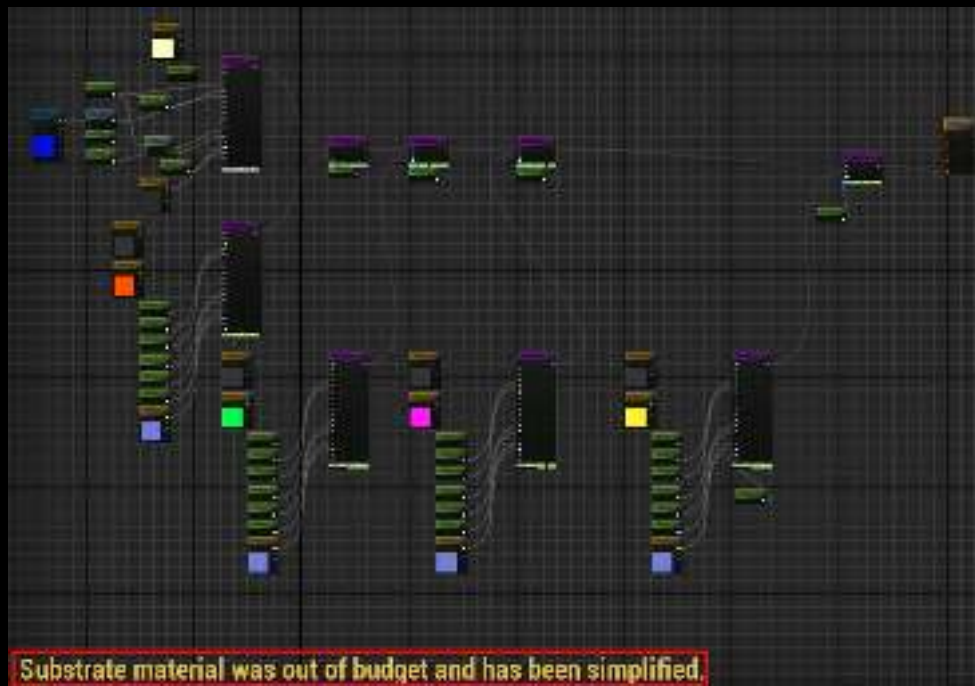


서브스트레이트 한계

서브스트레이트 한계

픽셀 당 바이트 수 제한

픽셀에서 너무 많은 메모리를 사용하면
자동으로 파라미터 블렌딩을 통해서
재질 간소화를 실행한다



```
r.Substrate=True  
r.Substrate.OpaqueMaterialRoughRefraction=True  
r.Substrate.BytesPerPixel=128
```

DefaultEngine.ini

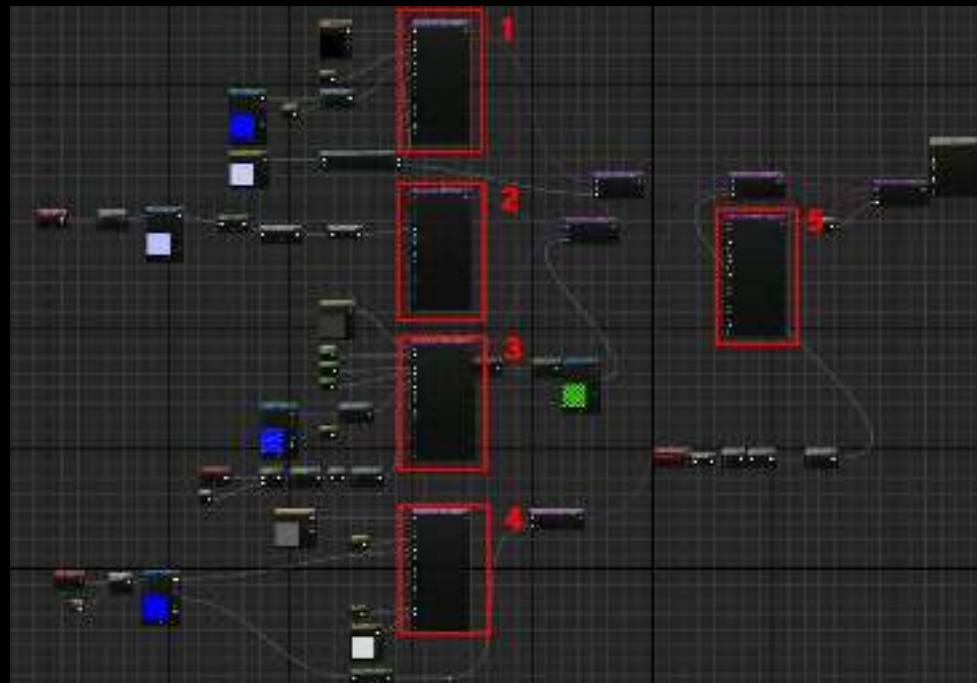
서브스트레이트 한계

고유한 노말 개수 제한

고유한 노말이 5개 이상일 때
컴파일 에러가 발생한다

슬랩이 5개라서가 아니고 슬랩에 연결된
고유한 노말이 5개라서

노말맵 텍스처를 공유한다면
컴파일 에러는 사라진다



[SMG] /Engine/Transient.PreviewMaterial1_7 [/Engine/Transient]: Substrate - Material has more unique normal/tangent basis than the allowed limit: 5/4.

서브스트레이트 한계

슬랩 개수 제한

슬랩이 8개를 넘어가면 컴파일 에러



⚠ [SM6] Material tries to register more BSDF than can be supported (9 > 8) See /Engine/Transient.M_Substrate_LimitationsC (asset: /Engine/Transient).

서브스트레이트 성능&디버깅

서브스트레이트 개발 기조

아티스트에게는 표현의 자유를

성능은 손쉽게 조절할 수 있도록



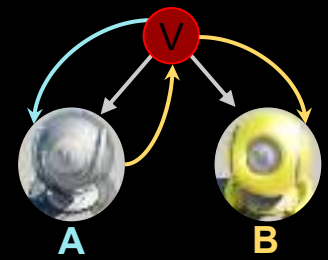
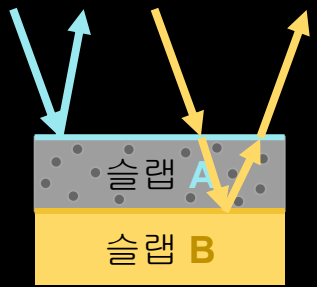
서브스트레이트 성능

서브스트레이트 트리

= 클로저 리스트

클로저 Closure

라이트 계산에 사용될 파라미터 묶음



클로저



+

클로저



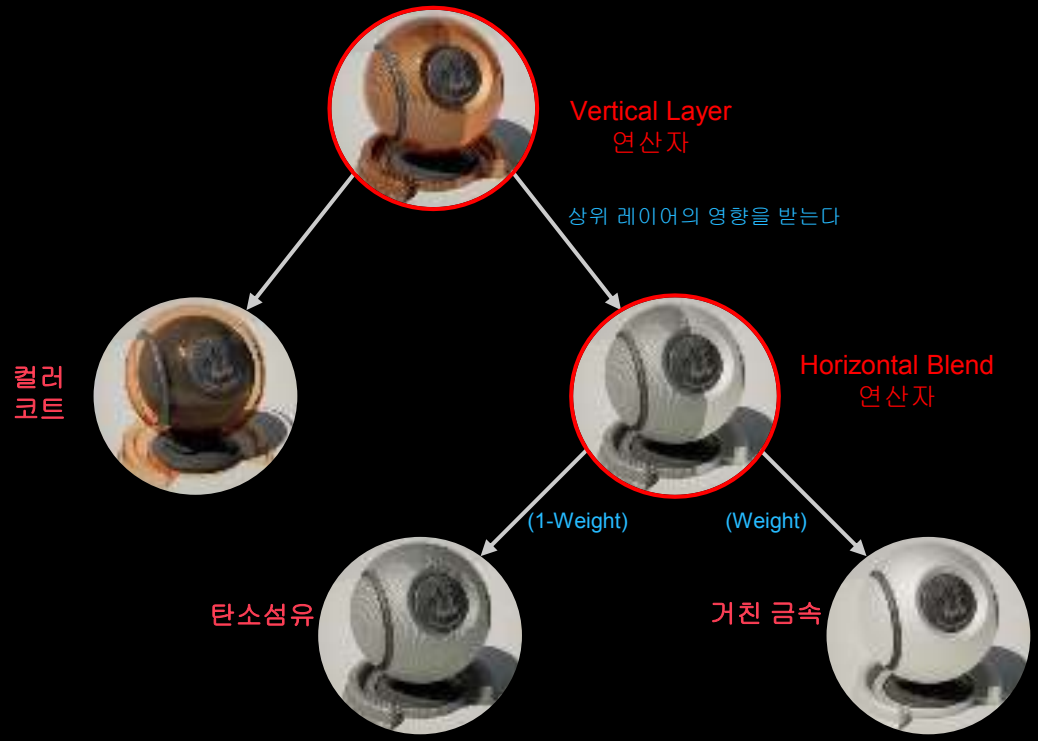
=



서브스트레이트 성능

컬러 결정

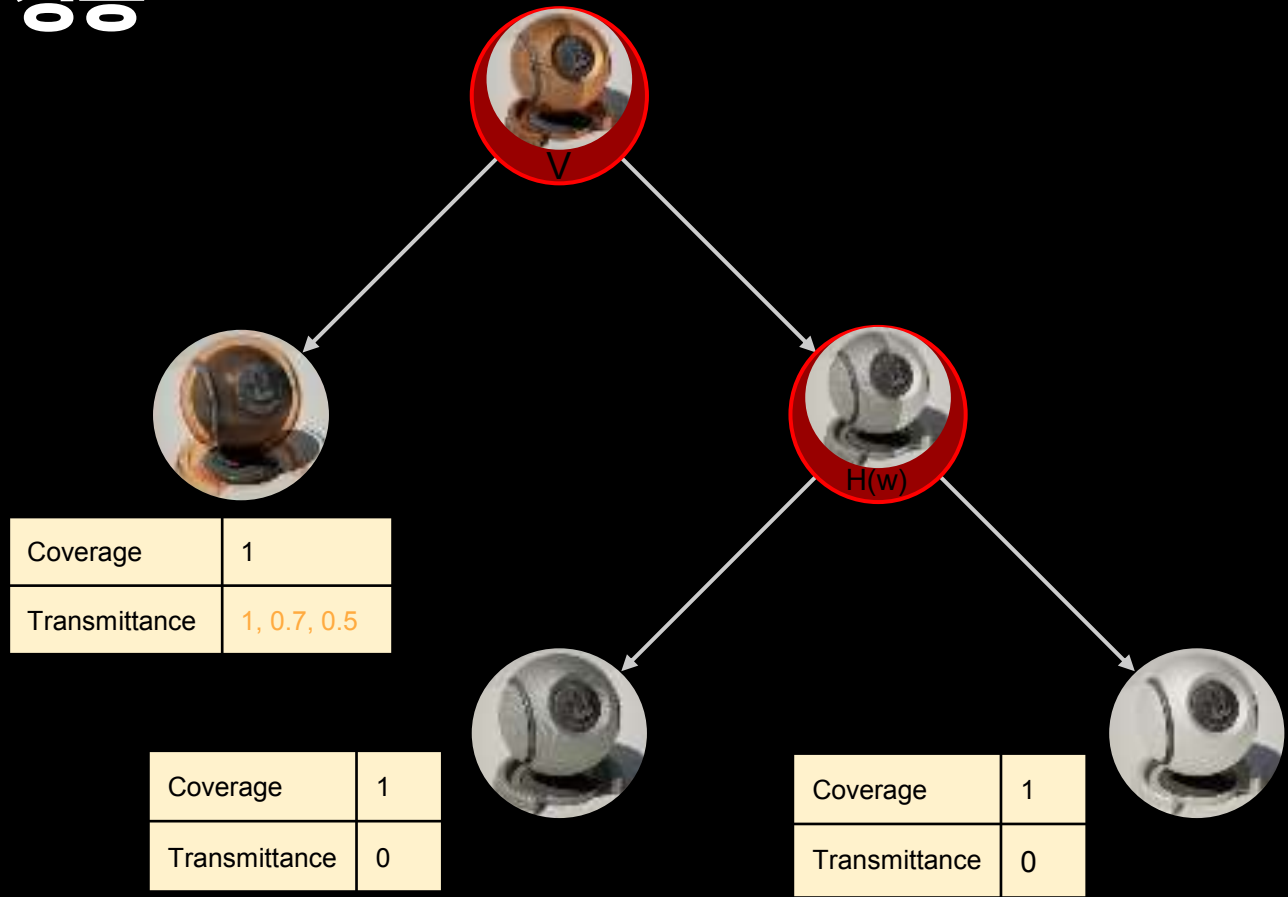
과정: 슬랩 → 클로저 → 루미넌스 컬러
최종 색 = 루미넌스 컬러의 합



서브스트레이트 성능

트리 탐색

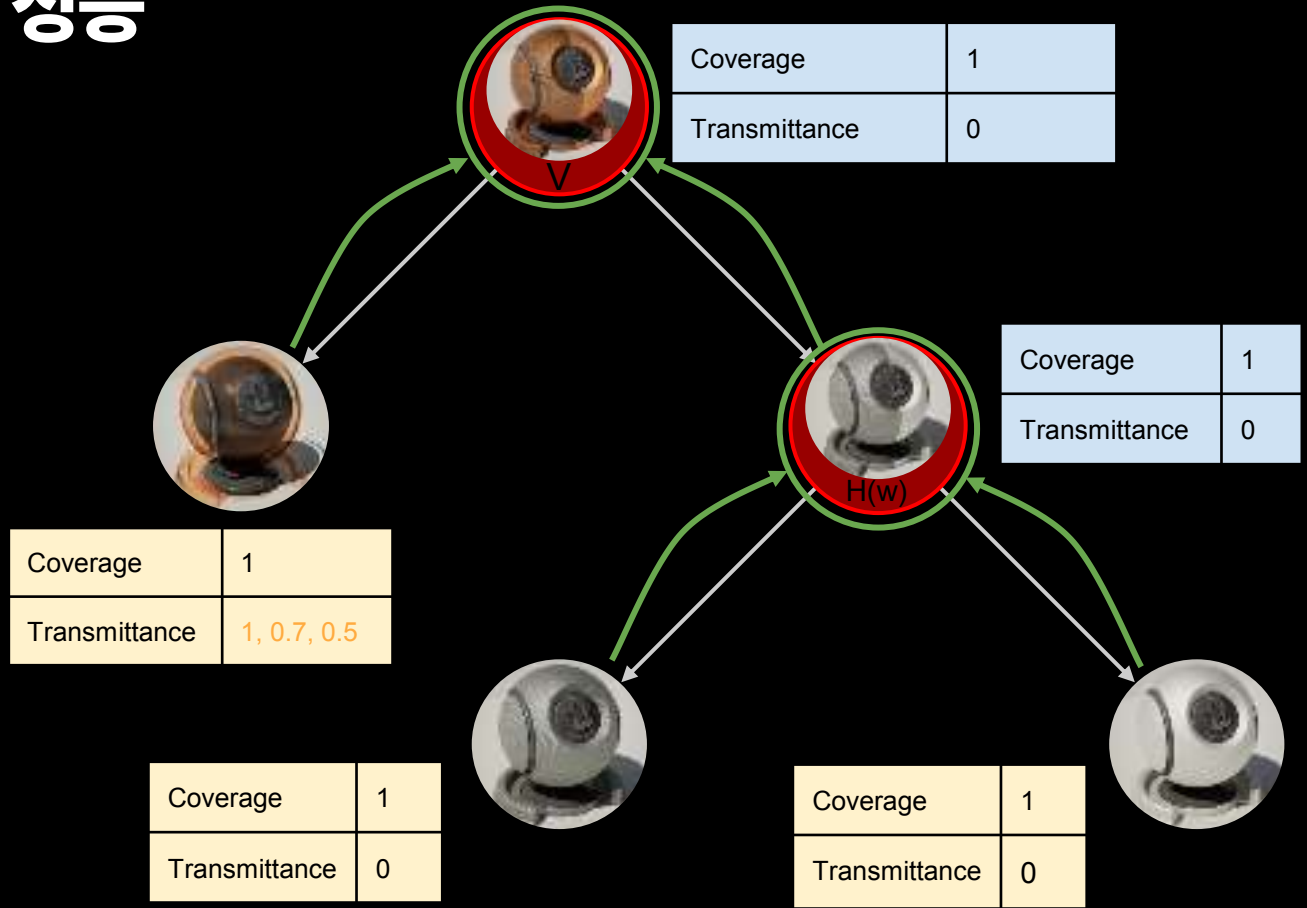
슬랩 커버리지/투과율 초기화



서브스트레이트 성능

트리 탐색

슬랩 커버리지/투과율 초기화
오퍼레이터 처리

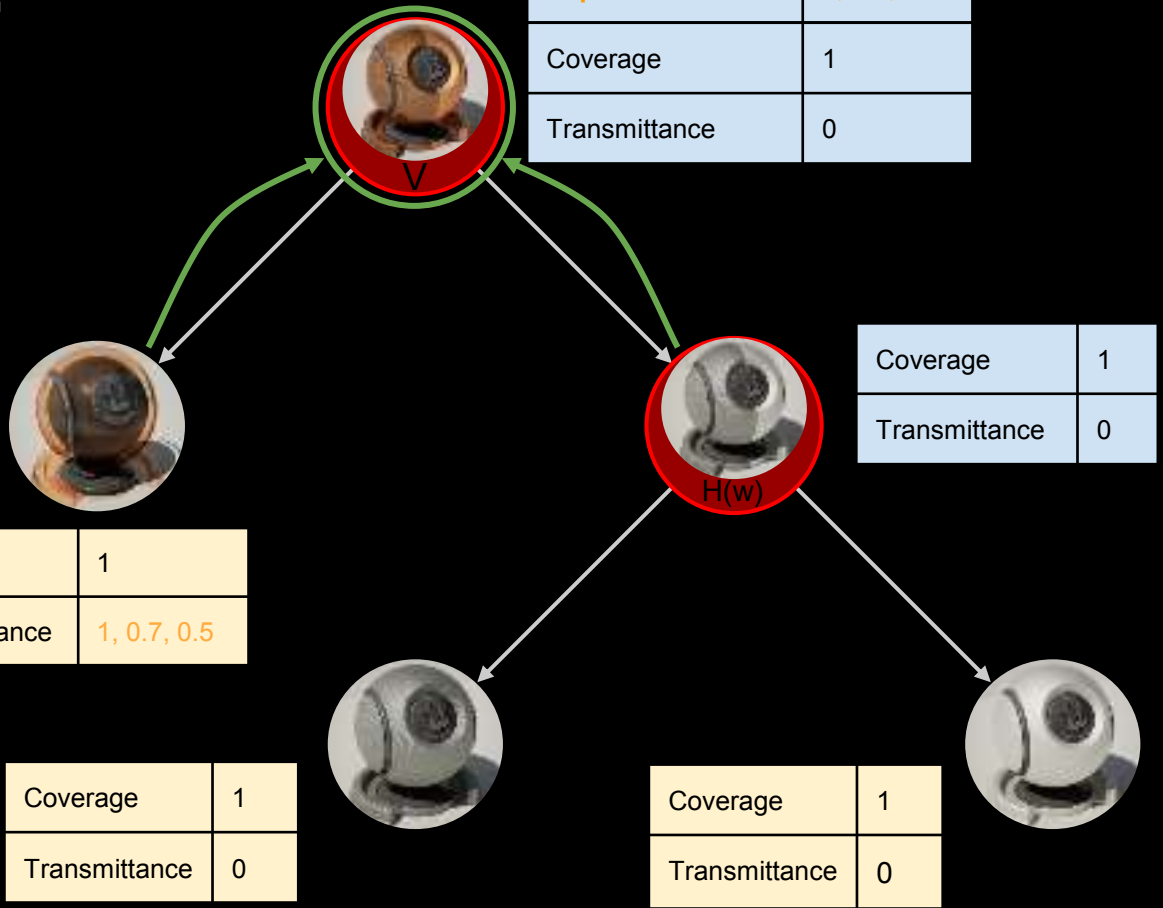


서브스트레이트 성능

트리 탐색

슬랩 커버리지/투과율 초기화
오퍼레이터 처리

TopCoverage	1
TopTransmittance	1, 0.7, 0.5
Coverage	1
Transmittance	0

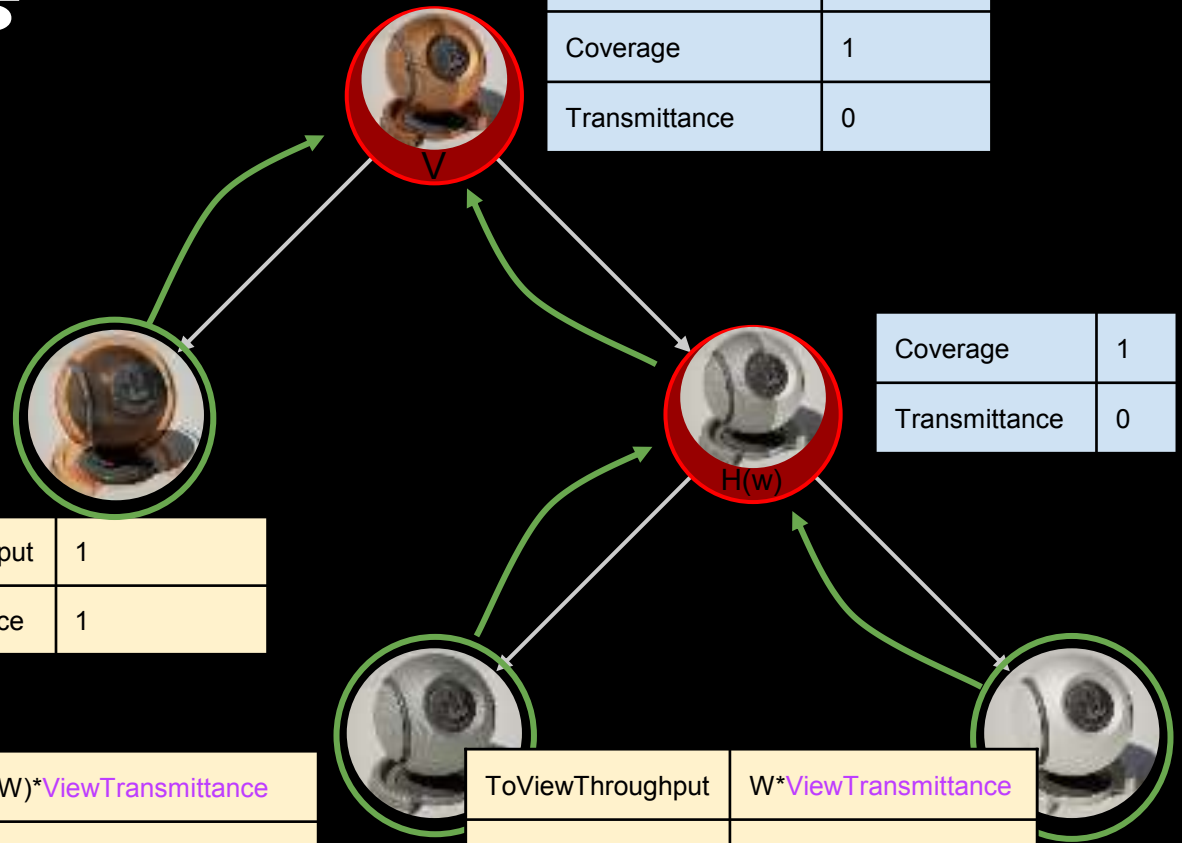


서브스트레이트 성능

트리 탐색

슬랩 커버리지/투과율 초기화
 오퍼레이터 처리
 슬랩 트리 워크

TopCoverage	1
TopTransmittance	1, 0.7, 0.5
Coverage	1
Transmittance	0



ToViewThroughput	1
TopTransmittance	1

Coverage	1
Transmittance	0

ToViewThroughput	$(1-W) \cdot \text{ViewTransmittance}$
TopTransmittance	TopTransmittance

ToViewThroughput	$W \cdot \text{ViewTransmittance}$
TopTransmittance	TopTransmittance

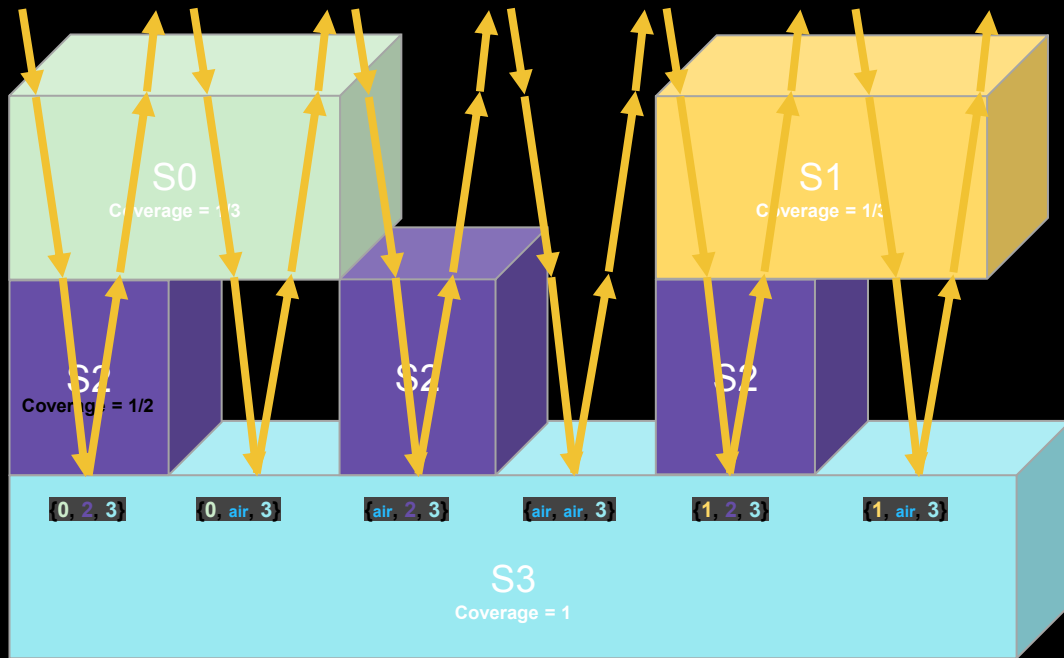
서브스트레이트 성능

멀티플랫폼 대응

= 부하를 조절할 수 있는 방안 필요

핵심은 클로저
복잡한 것을 단순하게

조합이 복잡해지는 것을 차단
트리 단순화



서브스트레이트 성능

멀티플랫폼 대응

= 부하를 조절할 수 있는 방안 필요

트리 단순화
= 파라미터 블렌딩

```
lerp (DiffuseA, DiffuseB, mix)  
lerp (F0A,      F0B,      mix)  
lerp (F90A,    F90B,    mix)  
normalize (lerp (NormalA, NormalB, mix))
```



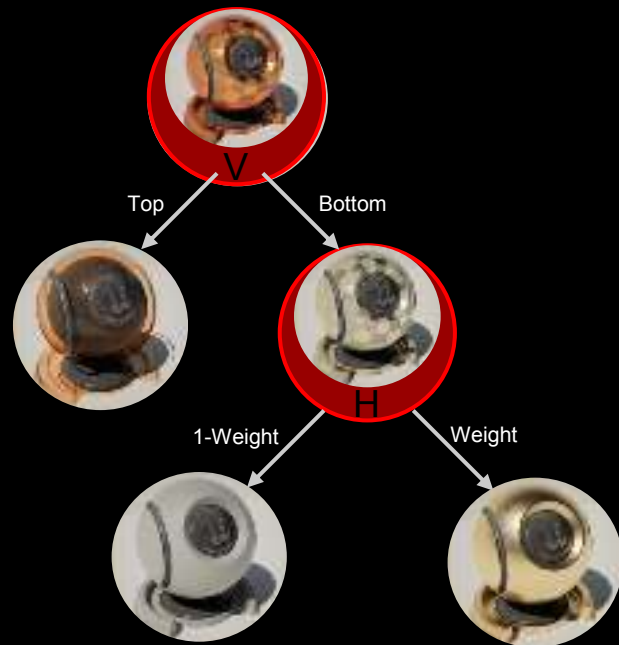
서브스트레이트 성능

멀티플랫폼 대응

= 부하를 조절할 수 있는 방안 필요

트리 단순화

= 파라미터 블렌딩



서브스트레이트 성능

멀티플랫폼 대응

= 부하를 조절할 수 있는 방안 필요

트리 단순화

= 파라미터 블렌딩

Vertical Layer, Horizontal Blend 노드 제공
비주얼 차이 발생

파라미터 블렌드로 부족하다?

SSS, Fuzz 등의 기능 끄기



서브스트레이트 성능

멀티플랫폼 대응

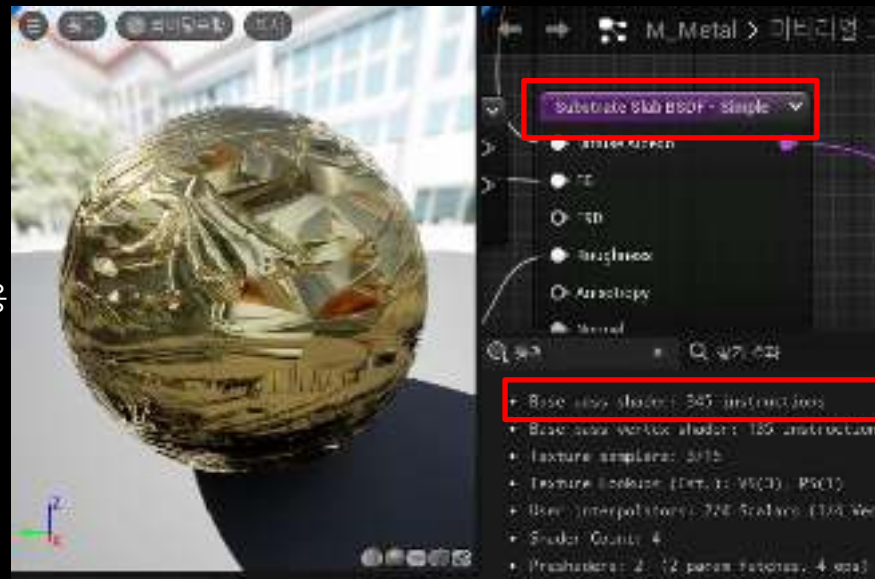
슬랩 별 복잡도

Simple = 간단한 재질 표현(Diffuse / F0 / Roughness만 사용)

Single = F90 / Fuzz / Second Roughness / MFP / SSS 사용

Complex = Anisotropic lighting 사용

Complex Special = 특수한 피쳐(Glint , Specular LUT) 사용



서브스트레이트 성능 - 디퍼드 셰이딩

레거시

서브스트레이트

		머티리얼	라이팅	머티리얼	라이팅
Simple	메탈	0.08	0.20	0.10	0.20
Single	클로스	0.09	0.21	0.12	0.25
	SSS	0.09	0.22	0.15	0.24
Complex	2 레이어			0.23	0.58
	3 레이어			0.31	0.79

PS5 1080P

서브스트레이트 성능 - 메모리

레거시

GBufferA (normal/obj data)	8.0	MB	// all GBuffer textures are RGBA8
GBufferB (Met/Spec/Rough/ShadId)	8.0	MB	
GBufferC (BaseColor/Occlusion)	8.0	MB	
GBufferD (CustomData)	8.0	MB	
TOTAL	32.0	MB	

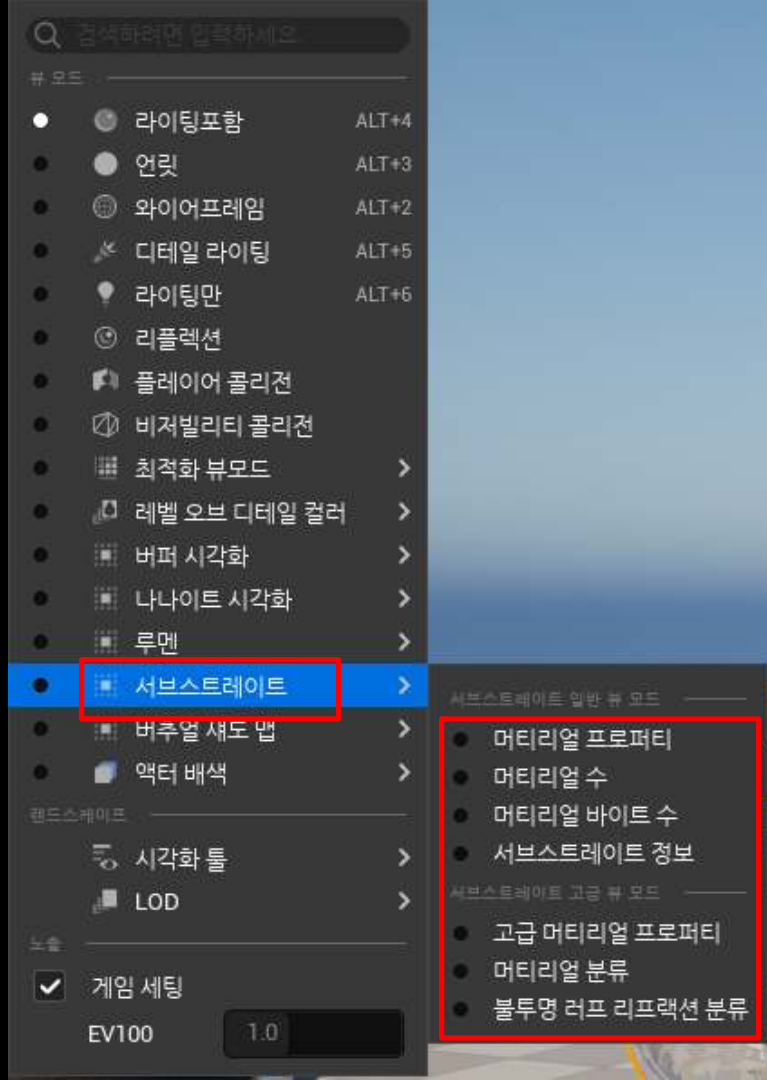
서브스트레이트

Substrate.Material	50.0	MB	// 4 UINT for SINGLE Closure + 2 SSS data => 6 UINT slices
Substrate.TopLayerTexture	8.0	MB	// Roughness / Normal
TOTAL	58.0	MB	

서브스트레이트 디버깅

디버깅 툴 제공

에디터 > 뷰포트 > 서브스트레이트 뷰모드



서브스트레이트 디버깅

디버깅 툴 제공

서브스트레이트 뷰모드 > 머티리얼 프로퍼티

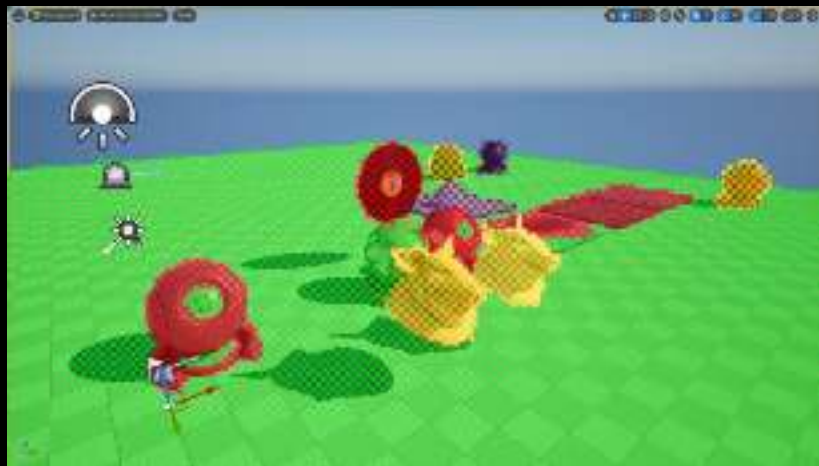
마우스 포인팅된 픽셀
최종 패키징된 클로저 정보 출력



서브스트레이트 디버깅

디버깅 툴 제공

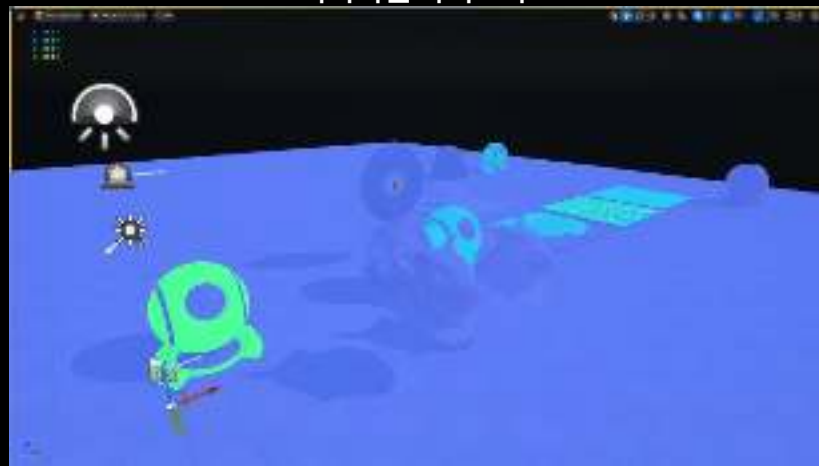
서브스트레이트 뷰모드 소개



머티리얼 분류 / 녹황적자 순서



머티리얼 바이트 수

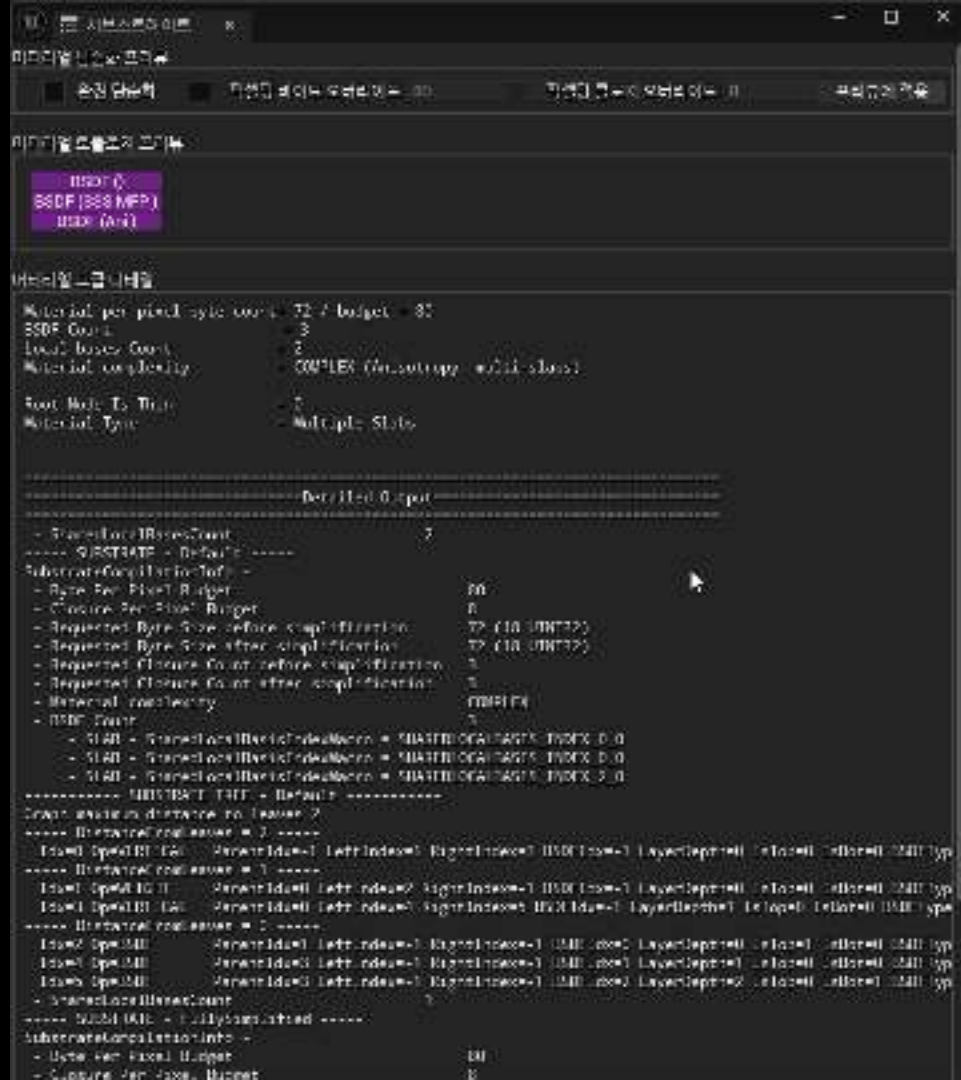


머티리얼 수

서브스트레이트 디버깅

디버깅 툴 제공

머터리얼 에디터 > 창 > 서브스트레이트
BSPD개수, 픽셀당 바이트 설정 등





감사합니다.

—에픽게임즈 코리아