

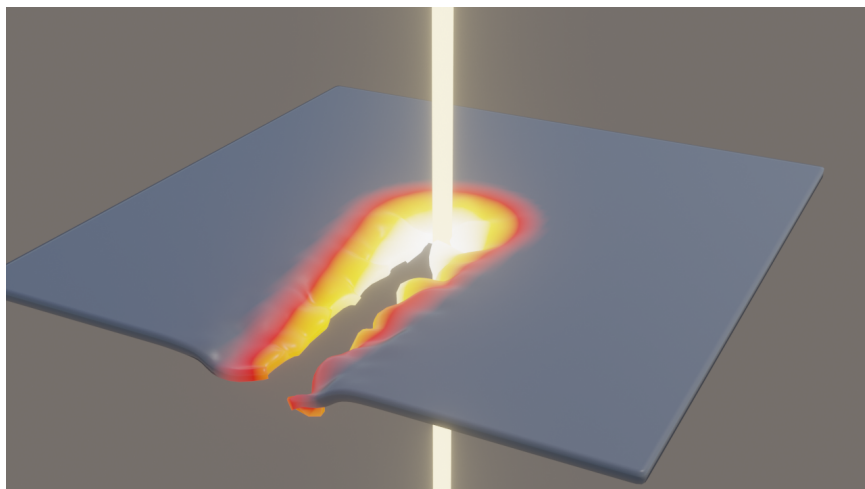
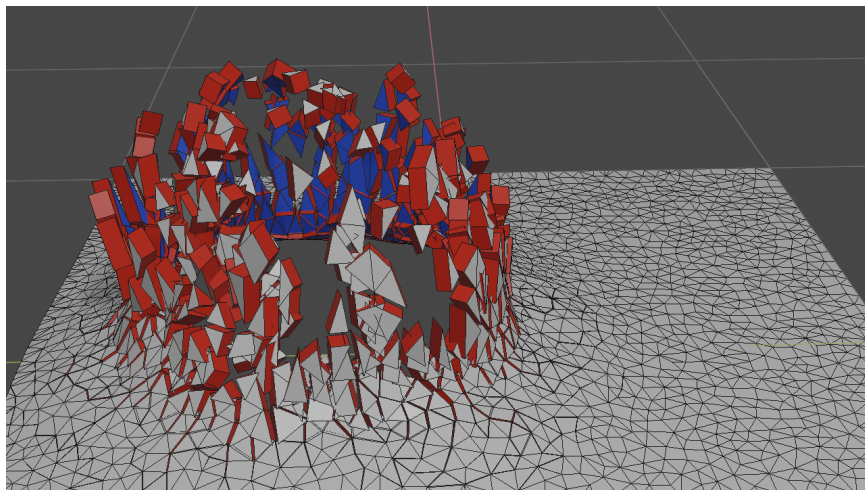
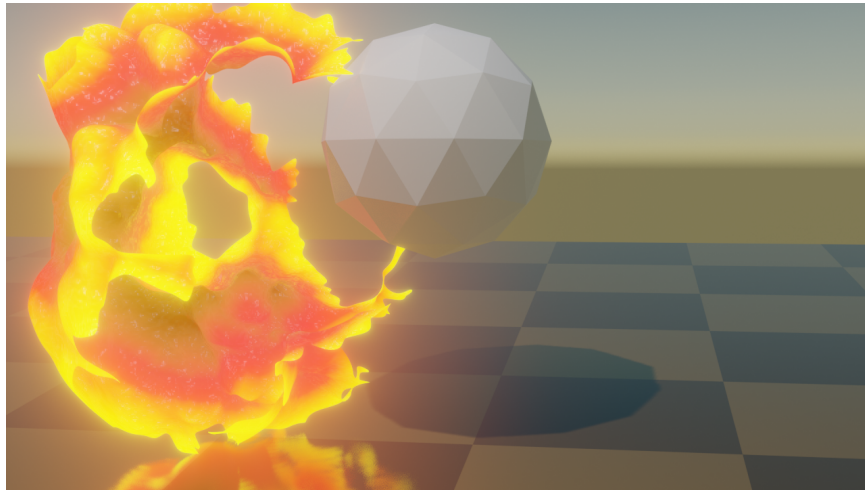
モディファイア芸でいろいろな効果を作ってみよう

Blender でのモディファイアのコンボの話 2.4版



Q@スタジオほぶり
@popqjp

作者 Twitter アカウント
[Q@スタジオほぶり](#)



目次

初めに ... 3	キャスト ... 38
モディファイアの基礎 ... 3	分割されたオブジェクトを、距離に応じて小さくして消す ... 39
サブディビジョンサーフィス ... 5	ワープ ... 40
ディスプレイス ... 6	はじけて消える球 ... 41
煙や雲のもくもくとした形状 1 ... 6	割れるようにはじける板 ... 44
一枚のポリゴンから地形作成 ... 7	ダイナミックペイント ... 46
リメッシュ ... 8	光線によって溶けながら切断される板 ... 48
辺分離 ... 9	溶接 ... 50
ソリッド化 ... 9	シンブル変形 ... 51
一枚のポリゴンを細かなパーツに分割 ... 10	360度曲げた円配置と、溶接 ... 52
デシメート (ポリゴン数削減) ... 11	飛び散る破片にツイスト変形をかけて竜巻的な表現 ... 53
一枚のポリゴンを、ややランダムなパーツに分割 ... 11	三角面化 ... 54
頂点ウェイト近傍 ... 13	ラブラシアン変形 ... 55
オブジェクトが接近した部分をへこませる ... 13	うごめく触手 ... 56
マスク ... 14	カーブ ... 58
オブジェクトが接近した部分のモデルを消す ... 15	波 ... 59
頂点ウェイト合成 ... 15	ビルド ... 60
パターンに従って格好よくモデルを消す ... 16	破片から元の形に再生する ... 60
煙や雲のもくもくとした形状 2 ... 17	ベベルと重み付き法線 ... 61
配列 ... 18	スキン ... 62
配列を使って増やしたオブジェクトの色を変える ... 19	パイプライン ... 63
テクスチャ違いのオブジェクトの配置 ... 20	メッシュのボリウム化 ... 64
インスタンス化 ... 20	ボリウム変形 ... 65
インスタンス化を使って増やし、ランダムに着色する ... 20	ボリウムのメッシュ化 ... 66
ややランダムな配置をインスタンス化で作成する ... 22	泡をまとわせる ... 66
シュリンクラップ ... 22	パーティクルインスタンス ... 67
立方体を床と天井に挟まれた空間にピッタリとはめ込む ... 23	モディファイアとジオメトリノード ... 69
頂点ウェイト近傍とシュリンクラップ ... 24	ジオメトリノードの基礎 ... 69
メッシュ変形 ... 25	ディスプレイス(Displace)のような変形 ... 70
スザンヌをピッタリとはめ込む ... 25	もくもくした形状 ... 71
ラティス ... 26	ジオメトリノードによるインスタンスの配置 ... 71
サーフェス変形 ... 26	配列(Array)モディファイアのようなノード ... 73
クロスシミュレーションからの変形 ... 27	アトリビュートの値を設定する ... 74
シュリンクラップ(Shrinkwrap)コンストレイント版 ... 28	頂点ウェイトを編集する ... 75
頂点ウェイト編集 ... 29	頂点ウェイトをシェーダーで使う ... 76
頂点ウェイト近傍による変形を波紋状にする ... 29	パーティクルとジオメトリノード ... 76
砂漠での爆発のようなエフェクト ... 30	動画とサンプル.blendファイル ... 78
UVワープ ... 31	終わりに ... 78
頂点ウェイト情報をUVに渡して、マテリアルで使用する ... 32	
フック ... 32	
フックによってソフトボディシミュレーションを制御 ... 34	
ソフトボディとインスタンス化 ... 35	
法線 (ノーマル) の編集 ... 36	
自動で開く多重扉 ... 37	
モディファイアのコピー ... 38	

モディファイア芸でいろいろな効果を作ってみよう

Blender でのモディファイアのコンボの話



Q@スタジオぽぷり
@popqip

作者 Twitter アカウント
[Q@スタジオぽぷり](#)

2019.07.30 ver 1.0	2021.12.03 ver 2.2
2020.07.12 ver 1.5	2022.03.19 ver 2.3
2020.10.09 ver 2.0	2023.09.17 ver 2.4
2021.04.10 ver 2.1	

初めに

Blenderの持つ重要な機能の一つにモディファイア(Modifier)があります。
正式にはモディファイアーと伸ばして訳されるのですが、著者の好みで短くモディファイアと表記します。

モディファイアが行うことは、基本的にオブジェクトの形状を変えることです。
オブジェクトの位置や回転の情報には影響を与えずに、その「形状」を操作します。
形状は変えずに、UV や 頂点ウェイト(Vertex Weight)に影響を与えるモディファイアもいくつかあります。

少し似ている機能として、オブジェクトコンストレイント(Object Constraint)がありますが、こちらはオブジェクトそのものが持つ位置や回転情報を操作することになります。

モディファイアの特長は

1. 元のメッシュのデータには影響を与えないので、何らかの操作をしてもモディファイアを除去すれば元に戻る。(非破壊編集)
 2. 複数のモディファイアを組み合わせることで、非常に複雑な操作を行うこともできる。
- などがあげられます。

一つ一つは単純な効果をもつモディファイアを複数組み合わせていろいろな効果をつくるのは、まるで頓智のようなものです。
うまいモディファイアの組み合わせで作ったエフェクトを、著者はよくモディファイア芸と呼んでいます。
本書では、今までに思いついたり偶然出来上がったモディファイア芸の例を幾つも挙げていくことをします。
特に、動く破片やエフェクトのような動きのあるモディファイア芸を中心に紹介します。

本書は、読者としてある程度 Blender の操作に慣れたブレンダー使い(少なくとも、使い始めたばかりで使い方を学ぶ段階は過ぎた)を想定しています。
基本的な操作の仕方は、必要に応じて記述しているところもありますが、だいたい説明をスルーしてすでに知っているものとしてますので注意してください。
本書の2.4版は基本的に Blender 3.6 に対応するように更新したものです。

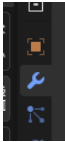
ジオメトリノード

blender 2.92 から 超強力なモディファイアとして、ジオメトリノードという機能が追加されました。
これは、ノードを組み合わせる様々な変形や複製を行うモディファイアです。
その性質上、ユーザーが上手くノードを作り上げれば、モディファイアの機能を代用することができる強力なモディファイアです。

ジオメトリノードは機能強化が進んでいて、blender 3.x 系統を経て blender 4.x のリリースの近い今となっては、主客が逆転した感があります。
「ユーザーがカスタム的に機能を編集することが出来る」ジオメトリノードに対して「予め用意された機能を使うための」モディファイア、というような関係です。
とはいっても、わざわざ難しい編集作業をしなくて良く手軽に組み合わせることができるという意味で、十分に既存のモディファイアにも意味があると言えるでしょう。
(ジオメトリノードでは対応できない機能を持つモディファイアだってあります)

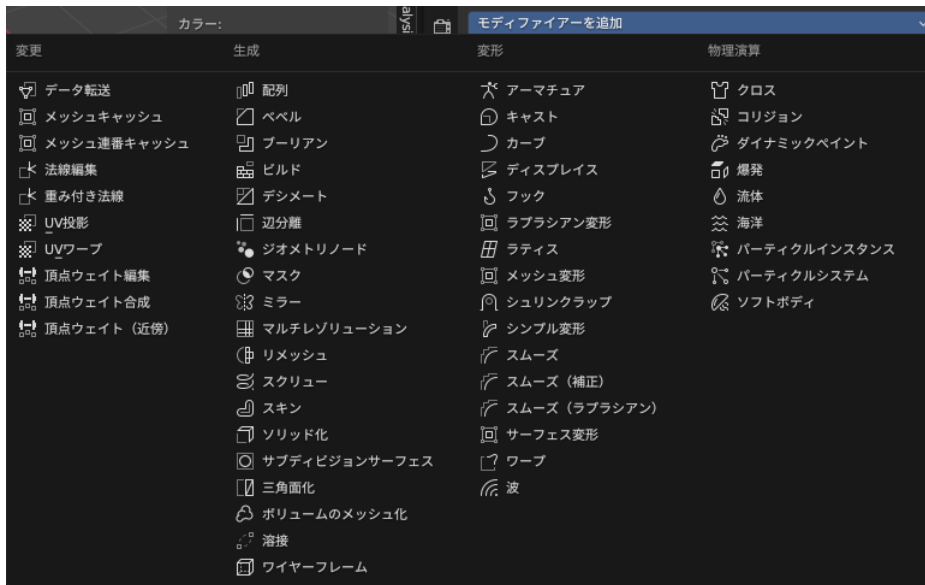
ジオメトリノードに関しては、それだけで複数冊(!)の解説書が作れるぐらいの機能があります。
ジオメトリノードに関しては本書では[ほんの触りのみ](#)を説明して、主にジオメトリノード以外のモディファイアについて説明をします。
ジオメトリノードについての説明の多くは、本書の2.4版で追加をしました。

モディファイアの基礎



モディファイアに関する操作は、プロパティパネルの中の青いスパナのアイコンのパネルで行います。

「モディファイアーを追加」ボタンからメニューを選択すると、オブジェクトに選択したモディファイアが追加されます。

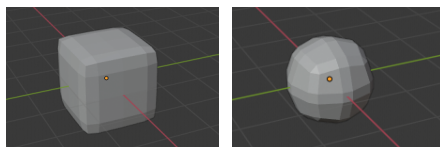


メニューの一覧は、大きく4つに分類されています。

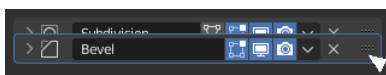
1. メッシュの形の変更は行わずに、そのほかのデータの変更を主に行うもの（変更）
2. メッシュの数を増やしたり減らしたりしながら形を変えるもの（生成）
3. メッシュの数は変えずに頂点の位置を変えるもの（変形）
4. そしてシミュレーションに関する物理演算です。

本書では、シミュレーション解説は範囲を超えるものとして、いくつかの例外を除き解説しません。
また、残りのモディファイアの中にも、いくつか扱っていないモディファイアがあります。
ジオメトリノードに関しては、非常に強力で複雑な機能ですが、本書の範囲を超えてしまうので、簡単にしか扱いません。

モディファイアは、処理速度とメモリの許す限り、いくらでも追加をすることができます。
上から順番に評価されるので、複数のモディファイアを利用するときは、その配置の順番也非常に重要です。



例えば、ベベルをしてからサブディビジョンサーフェスをするか、サブディビジョンサーフェスをしてからベベルをするか、という順番の違いで、最終的な形状は大きく変わります。



モディファイアの順番は、右側のボタンをマウスドラッグで操作することで入れ替えることができます。

基本的に、モディファイアによって形が変わったオブジェクトは、オブジェクトモードの時に表示されます。
エディットモードでは、変形前の元々の状態の形状が表示されます。
基本的に編集がしたいモードなので、そうならないと困りますね。
エディットモード中でモディファイアによって変化した形状を表示したい時も、表示方法を選択できます。



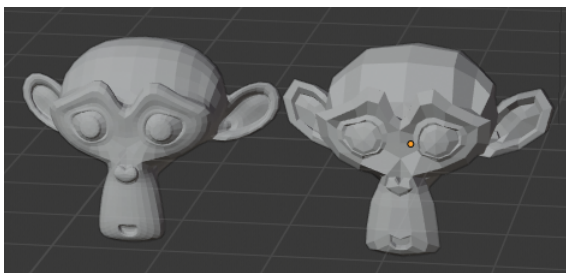
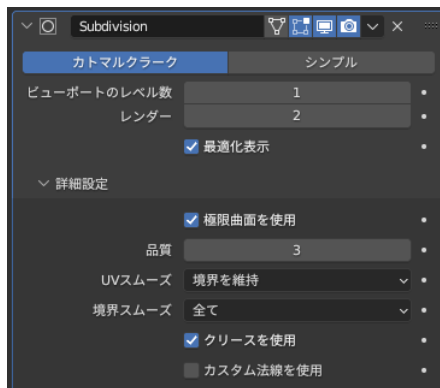
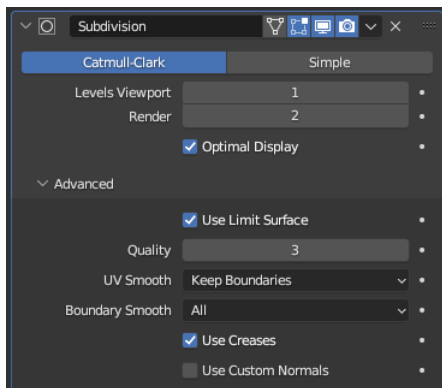
モディファイア上にある四角と三角のボタンはそれぞれ、

- ・変形した形状を同時に表示するモード
- ・さらに基本形状の頂点やメッシュが、最終的な形状のどの部分に対応しているのが表示するモード

となっています。
必要に応じてうまく表示を使い分けるようにします。

サブディビジョンサーフェス (Subdivision Surface)

最もよく使われるモディファイアの1つで、メッシュを細分化します。



カトマルクラーク(Catmul-Clark)はメッシュを滑らかな曲線になるように細分化し、シンプルは、形状を保ったままメッシュを細分化します。

単体ではシンプルは意味がないですが、細分化されたメッシュを別のモディファイアで利用できるので、こちらもよく使われます。

細分化のレベルは、レンダー時とビューポート時で違った設定にできます。レンダリングを行うときだけ、細かく細分化して高品質のモデルを使うことができます。

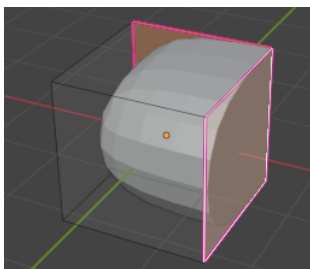
しかし、多数のモディファイアを組み合わせるときの、メッシュ数の違いから思わぬ挙動の違いが起きて、レンダリング時に全く違った見た目になってしまうことがあります。



レンダーとビューポートの細分化のレベルは、モディファイアを多数組み合わせるときは一緒にしておきます。

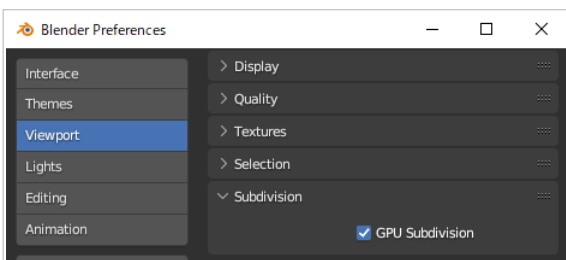


ショートカットの Ctrl-1 から Ctrl-5 は、サブディビジョンサーフェスに割り振られているので、キーボードで一瞬で細分化できます。



モデリングの時にエッジに設定できるクリーズ(Crease)の値は、カトマルクラーク細分化の時の滑らかさに影響します。

くっきり残したい辺でCreaseを1にすることで、一部をくっきり残したままで滑らかにすることができます。

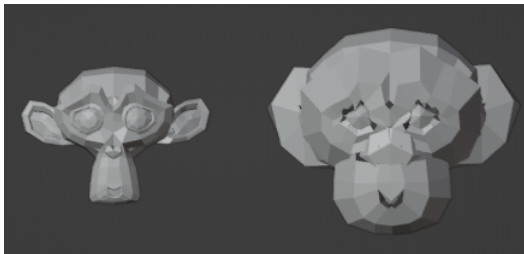
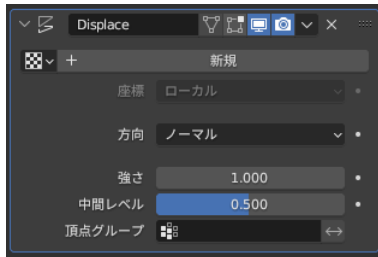
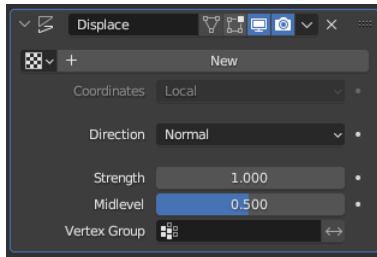


サブディビジョンサーフェスは、比較的重い処理で、特に分割のレベル数を上げると操作感がもっさりしてくる時があります。

Viewport(ビューポート)の設定で、この処理に GPU を使えるようにする設定があるので、(何か不都合が起きなければ)有効にしておくのがお勧めです。

ディスプレイス(Displace)

これも非常によく使われるモディファイアの1つで、各頂点の位置を移動させることができます。

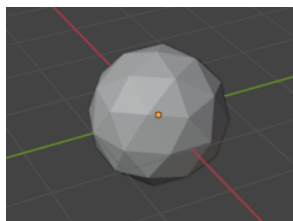


標準のスザンヌに、ディスプレイスをかけてみるとこのように膨れ上がります。
デフォルトでは、変形の方向がノーマル方向なので、表面が膨れ上がるように変形します。

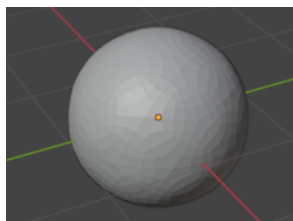
中間レベルは、使用するテクスチャの基準値をどこに取るかを決めます。
最初はテクスチャは未設定状態です。この時は、変形の度合として1を設定したものと扱われます。
すると、中間レベルは最初は 0.5 なので、その差の $0.5 \times \text{強さ } 1.0 = 0.5$ だけ各頂点が膨れる方向に移動したことになります。

この時テクスチャとして新規テクスチャを作った直後は、最初の真っ黒いテクスチャなので、0 として扱われます。
中間レベルとの差は -0.5 になるので、膨れるのではなくやや細る方向にスザンヌが変形することになります。
テクスチャを適切に利用すると、膨れる形状をコントロールできます。

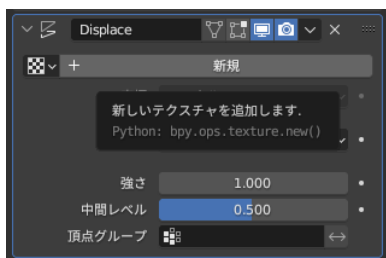
煙や雲のもくもくとした形状 1



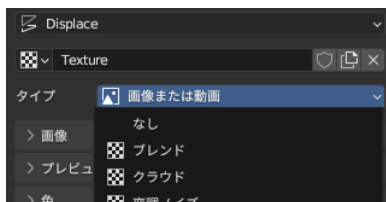
例として、球からもくもくとした形状を作ってみます。
まずICO球を作成します



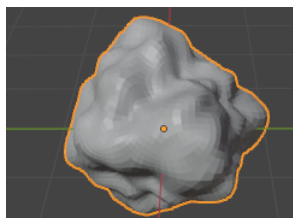
サブディビジョンサーフェスを追加して、細分化レベルを2か3にして十分に細かくします。



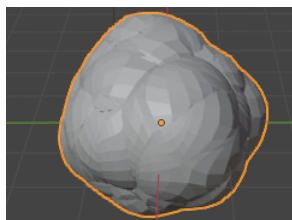
ディスプレイスを追加して、新規テクスチャを作成します。
その後、右にある設定ボタンを押して、テクスチャの設定に入ります。



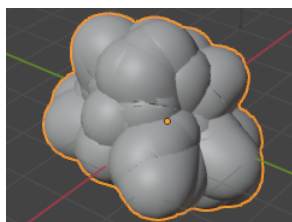
テクスチャのタイプをクラウドに設定します。
一様に膨らんでいた球が、テクスチャの模様に応じて膨らむようになり、凸凹に変形します。
そのままだと凹凸が細かすぎるので、クラウドのサイズ設定を1にします。



不規則な形で凹凸が付きました。
ここがミソですが、もう一つディスプレイスを追加します。

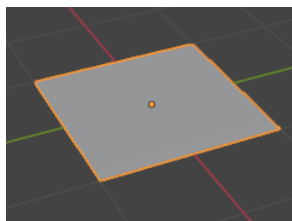


すると、いちど変形した形に対して、膨らませる方向にさらに変形が入ります。その結果、球が連なったようなもくもくした形状になります。
パラメータを少し変えて、調整をしてみましょう。

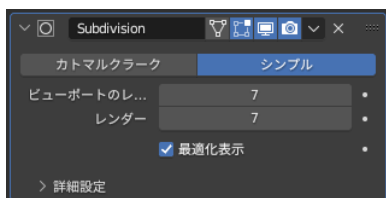


細分化を5まで増やし、クラウドのサイズ設定を0.75、最初のディスプレイスの強さを4、2番目のディスプレイスの強さを1.2にしました。
スムーズシェーディングにすることで、球が連なるスタイルでの爆発の瞬間をとらえたような形状になりました。

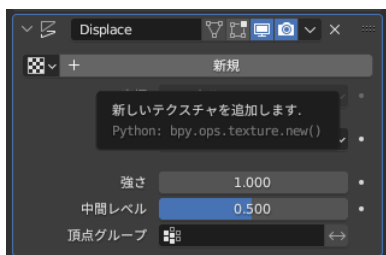
一枚のポリゴンから地形作成



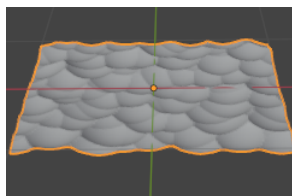
一枚のポリゴンを分割して変形して、地形を作成します。
まず平面を作成します。



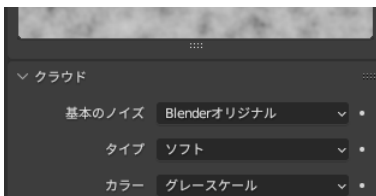
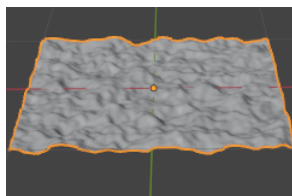
サブディビジョンサーフェスを追加して、細分化します。
元々のポリゴンが粗いので、地形の細かさを確保するために細分化のレベルを7（サブディビジョンとしてはかなり大きい値）にします。
分割の仕方はシンプルにしておきます。



ディスプレイスを追加してテクスチャを設定します。
そのままだとおそらく変形が強すぎるので、強さを適切な値に設定します。



例としてポロノイだと砂漠の風紋のような形状になります。



先ほども使ったクラウドだと凸凹な地形になります。



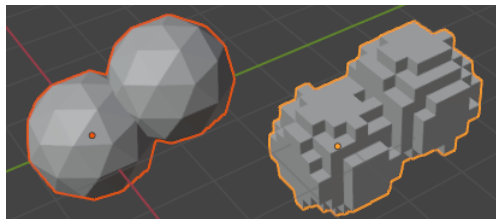
ポロノイのデフォルト設定の「実際の距離」で変形すると、すり鉢状の地形の底がとんがった形になります。
地形の底を滑らかにするには、二乗距離を使います。

リメッシュ(Remesh)

このモディファイアは、ポリゴンの形状を新たに作成します。

大きく分けて、ブロック系の処理と、ボクセル処理の2通りに分かれます。

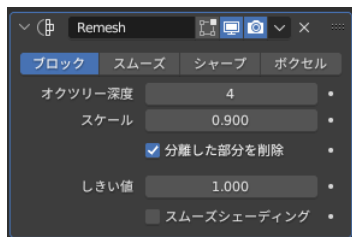
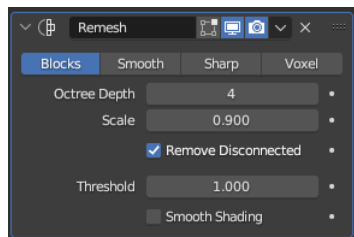
ブロック系の処理は、マインクラフトを思い起こさせるブロック状の形状を作成します。



いわゆる「閉じたメッシュ」つまり、メッシュに隙間が無くメッシュ同士が交錯していない場合は、綺麗にブロック状の形状になります。

閉じていないメッシュや、メッシュが交錯したような場合は、うまく行くとは限りません。

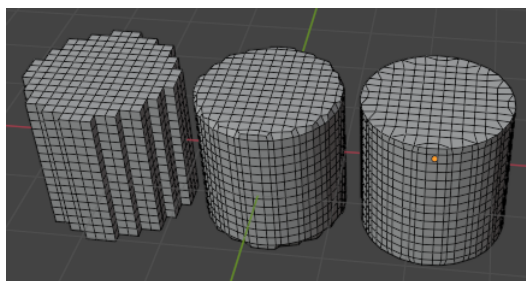
左の図で、よく見るとところどころ隙間があるのを見ると分かるのですが、詰まったブロックではなく殻のようなブロックになったりします。



ブロックの細かさは、オクツリー深度 (Octree Depth) で調整ができます。

また、スケール (Scale)を使ってさらに細かい調整ができます。

スケールはブロックの境界と、メッシュの形状が都合の悪いような位置関係になった場合などの微調整に使えます。



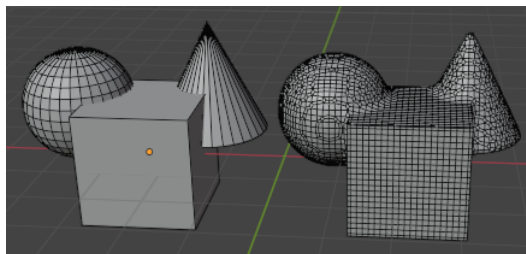
ブロックモードは、明らかに演出としてカクカクした形状を作りますが、スムーズ (Smooth) とシャープ (Sharp) は、形状を残したままでメッシュを荒く (細かく) するようなリメッシュ処理用のモードです。

シャープの方は、角の部分の情報をできるだけ残すようになっています。

左の図のように、円柱を元にしたリメッシュのメッシュを見てみると違いが分かります。

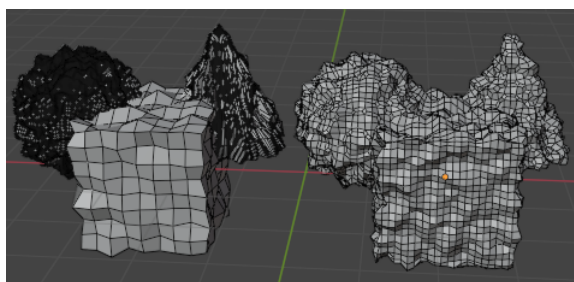
円柱のような形状なら、シャープモードが角の残った綺麗なリメッシュをしてくれています。

ボクセルモードはより新しい手法で、閉じた面の内部と外部をよりの確に判定してくれます。



図のように、お互いにめり込んでいるようなメッシュは、溶接したようにくっついた1つの塊になります。

頂点の密度に差があるような形状も、ほぼ様な頂点の密度になるので、別の処理がしやすくなるなどの利点があります。



先ほどの、サブディビジョンサーフィスとディスプレイスを使ってもくもくさせる変形で比較してみます。

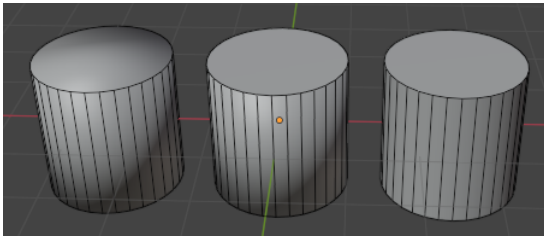
頂点の密度に差があったり、円錐のように頂点の配置が偏っている場合は、変形があまりきれいではありません。

リメッシュを利用することで、きれいな形に変形することができます。

辺分離(Edge Split)

このモディファイアは、一定の条件下でつながっているポリゴンの辺を分離してバラバラのポリゴンにする機能があります。

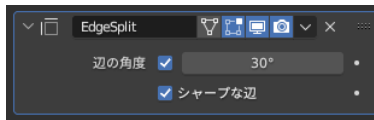
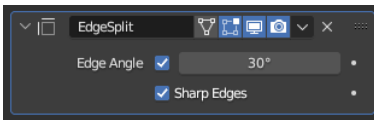
単独での使い方としては、円柱などのように「滑らかにつながって欲しい面ときっぱりと直角などになってほしい辺が混ざっているような形状」をうまく表示する、などがあります。



円柱を丸ごとスムーズシェーディングすると、上下の面に引きずられて陰影が球のものに近くなってしまいます。(左)

上下の面だけフラットシェーディングにしても、上下の面はフラットになりますが、側面はそのままです。(中)

辺分離によって、上下の面と側面を完全に分離することで、あるべき円柱の表示になりました(右)



Edge Split で分離する辺の指定は、角度で指定して自動で判定をするか、もしくは直接メッシュの辺に対してシャープ設定を与えることもできます。

円柱を綺麗に表示するだけであれば、辺分離でポリゴンを分割しなくても Auto Smooth を有効にすることで、スムーズシェーディングとフラットシェーディングを使い分けて綺麗に表示できます。

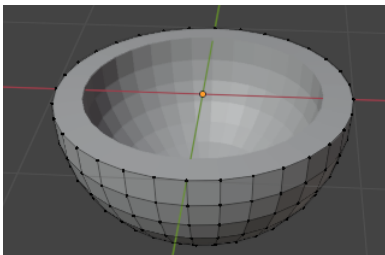
しかし後で出てくるような、他のモディファイアなどと組み合わせると、ポリゴン片をバラバラにするようなエフェクトには、辺分離が不可欠になります。



角度を0度に設定すると、すべての辺を分離することになるので、ポリゴンモデルを完全にバラバラにすることができます。

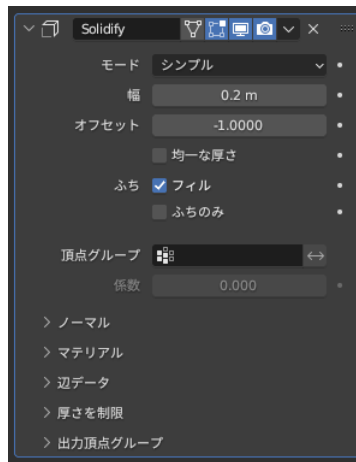
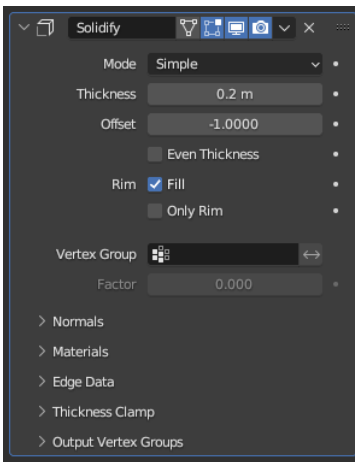
ソリッド化(Solidify)

このモディファイアは、一枚の厚みのないポリゴンに厚みを付けることができます。



球の上を削除した半球に対して、ソリッド化を追加してみました。

このように厚さがなかった半球が、厚みのあるオブジェクトになります。



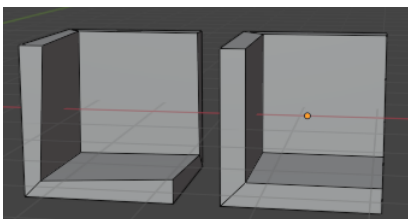
幅のパラメーターで厚さを指定します。

オフセットの値によって、ポリゴンの表側が膨らむように厚みを持たせるか、裏側が膨らむように厚みを持たせるかが変更できます。

ソリッド化のモードには、シンプル(Simple)と複雑(Complex)があり、「複雑」モードはシンプルな計算方法では破綻するより複雑な形状にも対応するためのモードです。

ただし、「複雑」モードはより低速になる欠点もあります。

このモディファイアはオプションが多いので、全体を把握するには[公式のマニュアル](#)をチェックして欲しいのですが、重要と思われる項目だけ確認してみます。



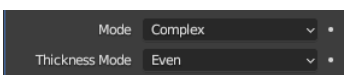
ソリッド化を、角のはっきりした面に対して行くと、左の図のように厚みに差が出ます。

これは、厚みイコール元の頂点からの距離と考えているためで、かどから斜めに新しい頂点を生成すると、斜め45度傾いている分だけ、本当の厚みとしては薄くなってしまいますからです。

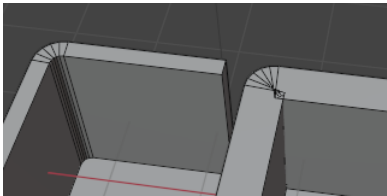
似たような問題は、カーブオブジェクトを太さを持ったメッシュに変換する時にも発生したりします。



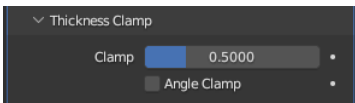
Even Thickness(均一な厚さ)を有効にすると、こうしたかどの斜めになる効果を考慮してソリッド化をします。



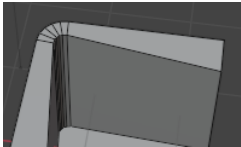
Complex (複雑) モードの時には、Even(均一)か、Constraint(コンストレイント)モードにすると、厚み一定のソリッド化になります。



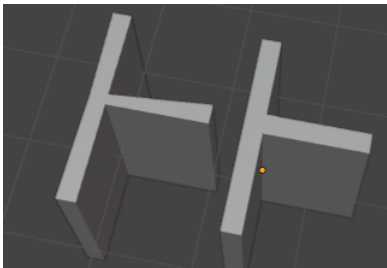
また、曲がった面に内側に向かって厚みを付けて、ある程度以上厚くすると、移動した頂点の位置関係が逆転して、形状が破綻してしまいます。(右側)



Thickness Clamp(厚さを制限)を0以上に指定すると、そのようなめり込みを防ぐように曲がった部分の厚みに制限がかかります。

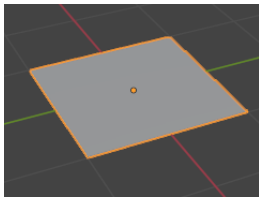


ただし、厚みが変わってしまうので断面の形状は美しくはなりません。平面と曲面の境に頂点(辺)を増やすなどで、もう少し形を整えることはできるのですが…こういっためり込みが起きないような形状で使うように心がけたいところです。

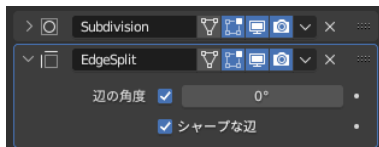


シンプル(Simple)と複雑(Complex)の計算モデルの違いですが、一番大きな違いは、面が分岐するような構造に対応しているかどうかです。このように(断面が)T字に分岐しているような面に厚みを持たせた際に、Complex モデル(右)では破綻せずに厚みを持たせることが出来ます。

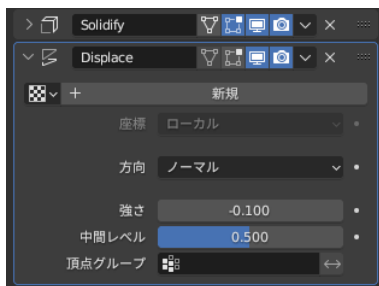
一枚のポリゴンを細かなパーツに分割



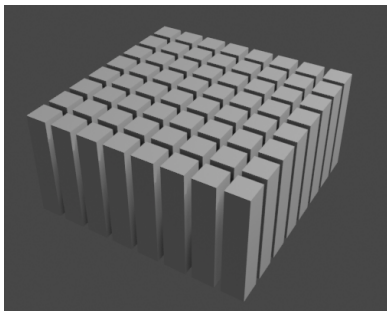
平面を作成します



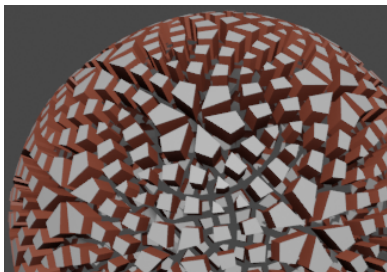
サブディビジョンサーフェスを追加して、細分化します。そして、辺分離を追加して、分離角度を0に設定します。これにより、すべてのポリゴンがバラバラに分割されます。



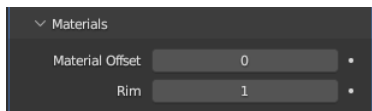
さらにソリッド化を追加して厚みを持たせます。その状態ではまだすべてのパーツが密着していて、バラバラに分割した効果が見えません。そこで、ディスプレイスを追加して、強さをマイナスにして、すこし全体が痩せるように変形します。



一枚のポリゴンがバラバラのパーツに分解されました。



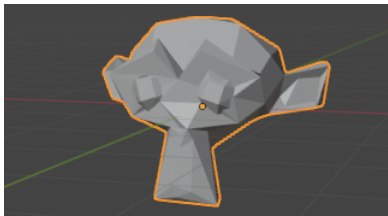
板ではなく、ICO球からポリゴンを分割することで、このような形状も作成できます。
ソリッド化の Material(マテリアル)設定で、縁のマテリアルを別のマテリアルにして、割れ目は別の色で表示することもできます。



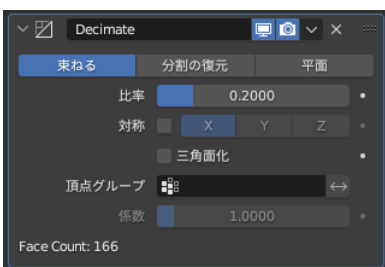
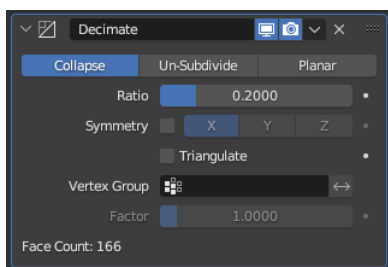
ソリッド化で作られた面や、縁(Rim)部分のマテリアルインデックスが、与えた数値だけずれるという仕組みです。

デシメート(Decimate)、ポリゴン数削減

このモディファイアは、できるだけ形状を保ったままポリゴン数を削減します。



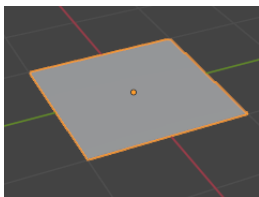
スザンヌのポリゴン数を20%まで削減してみました。



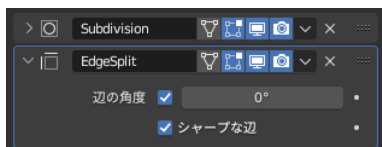
Collapse(束ねる)モードでポリゴン数を減らすときは、基本的に平らな部分のポリゴンからまとめて、平らでは無い場所ではできるだけそのままに残すようにして形を保ちます。

優先順位が同じようなポリゴンが削減されるときは、ランダムに近い形でポリゴンがまとめられていくので、以下のようなちょっとした応用ができます。

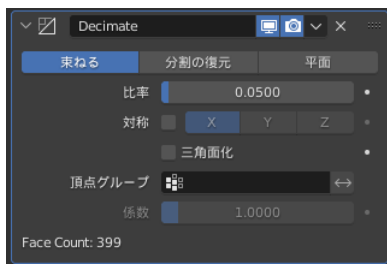
一枚のポリゴンを、ややランダムなパーツに分割



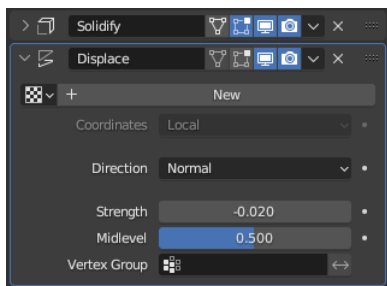
平面を作成します



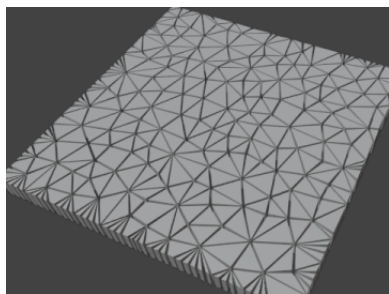
先ほどと同じようにサブディビジョンサーフェスを追加して、細分化し、辺分離することでバラバラにします。分割数は6程度にしてみます。



ここで、サブディビジョンサーフェスと辺分離の間に、デシメートを挟み込みます。すると、一様に細かく分割されていたポリゴンが、あちこちでまとめられて不規則な形のポリゴンになります。



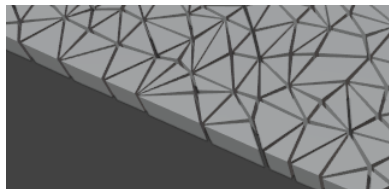
ソリッド化を追加して厚みを持たせ、ディスプレイによって少し痩せさせることで、境界をはっきりさせます。



規則正しいグリッド状ではなく、かなり不規則な三角柱の集合体として平面が分割されました。

この状態で、「サブディビジョンサーフェス」「ポリゴン数削減」「デシメート」「ソリッド化」「ディスプレイ」の5つのモディファイアを利用しています。

完全に平面の板を Decimate すると、縁の処理がやや気になる形状になります。



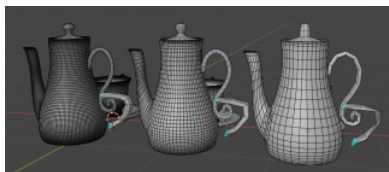
デシメートのモディファイアをかける前にほんの少し Displace モディファイアなどを使ってをかけて歪めると、縁の処理があまり気にならなくなります。

(試してみたところ、強さ0.0001とかでも大丈夫です。完全な平面でさえなくなれば良いみたいです)

このサンプルを Fracture.blend として同封しました。

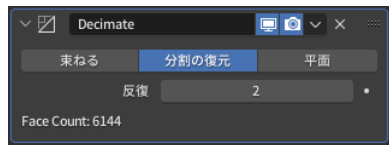


デシメートには面白い機能として、サブディビジョンサーフェスをかけたメッシュを、元に戻す機能があります。



モードを分割の復元にして、サブディビジョンサーフェスをかけたメッシュに使うと、元の形状にかなり近い状態になります。(四角メッシュで綺麗に形成されている場合にのみ使えます)

フリー素材などをダウンロードしたときに、形は良いのだけれどもサブディビジョンをかけすぎて必要以上の頂点数になっている、というような場合の頂点数削減に使えます。



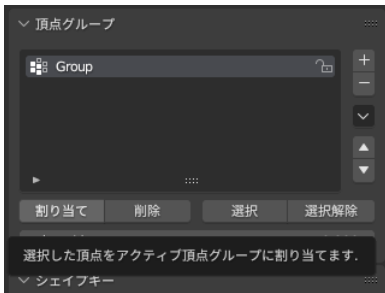
Un-Subdivide(分割の復元)モードで使用します。

サブディビジョンの1回分を戻すのに反復(Iterations)の回数を2使うので、通常は偶数の数値を入力します。

(奇数にすると、メッシュに斜めに線が入るので、そういう効果が欲しい時には使えます)

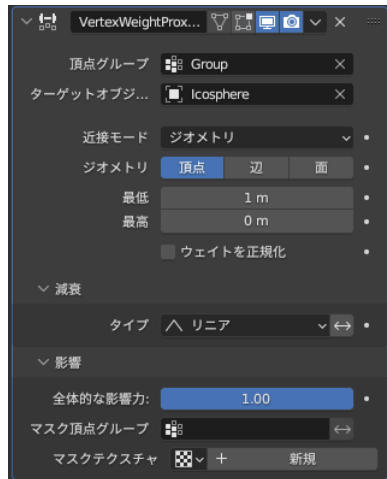
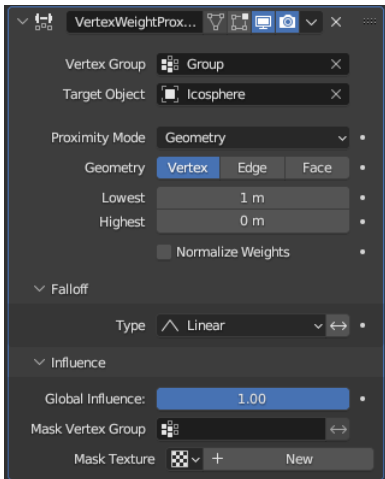
頂点ウェイト(近傍) (Vertex Weight Proximity)

このモディファイアは、他のオブジェクトとの距離から頂点ウェイトの値を動的に設定します。
動くオブジェクトと組み合わせてアニメーション効果にも使える、隠れた優れモノと言えます。
設定が多く、使い方が若干ややこしいので簡単に解説をします。



操作の対象になる頂点ウェイトがそもそも設定されていないと機能しません。
頂点グループの追加をしただけでは駄目なので、割り当てを忘れないようにします。

! 初期のウェイトは何を割り当ててもよいのですが、
0を割り当てるとサブディビジョンサーフェスによってメッシュを細分化した際に、
増えた頂点がウェイト未割り当ての状態となってしまいます。
細分化する場合は、ウェイトに0が必要な時も、0.001といった値を割り当てておくのが良いでしょう。



操作の対象となる頂点グループとターゲットオブジェクトを設定します。

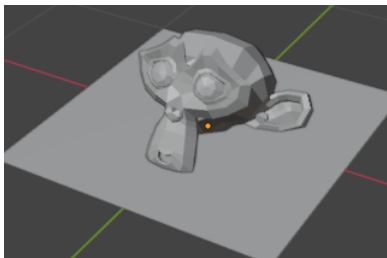
距離の設定は「オブジェクト」にすると、オブジェクトどうしの距離に対応した値が一樣に割り当てられます。
「ジオメトリ」にすると、対象のターゲットの形状に応じて頂点ごとにウェイトが割り当てられるので、より細かな制御ができます。

💡 初期の設定では、ターゲットに近い0m地点でウェイトが0（最低）で、1m離れるとウェイトが1（最高）になる設定で、名前から受ける印象と逆です。
近いほど影響が強いようにするには、左図のように逆の設定になるのに注意です。

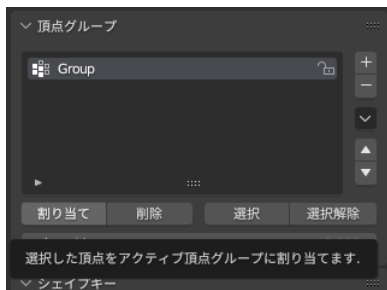
頂点ウェイトが変化するだけなので、単独では見た目は何も変わりません。
そこで、さっそく他のモディファイアと組み合わせてみます。

オブジェクトが接近した部分をへこませる

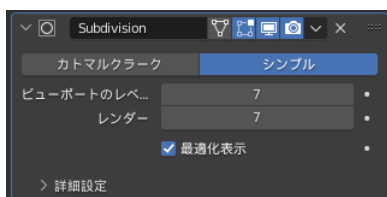
スザンヌが近づいた部分をへこませてみます。



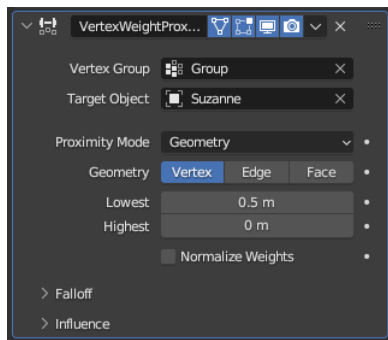
平面と、その近くにターゲットにするためのスザンヌを配置します。



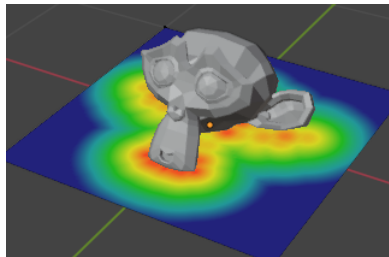
平面に頂点グループを追加して、頂点ウェイトを設定します。0以外を割り当てます。



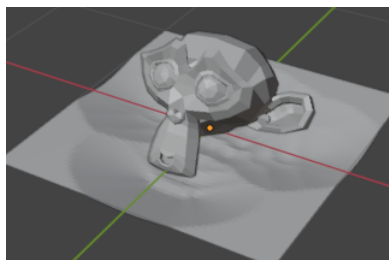
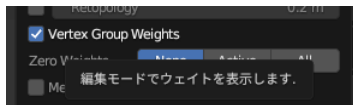
サブディビジョンサーフェスで分割します。分割数は多めにとっておきます。



ターゲットをスザンヌにして頂点ウェイト近傍を設定します。
ジオメトリに反応するようにして、近いほど大きな値になるように最高と最低の値を決めます。



頂点ウェイトが変化した結果は、頂点ウェイト編集画面などで見るができます。
他にも、[編集時にモディファイアの影響後のメッシュを表示するようにして](#)、オーバーレイオプションで頂点ウェイト表示を有効にすると、
編集モードでも、ウェイトが変化しているのを見ることがもできます。



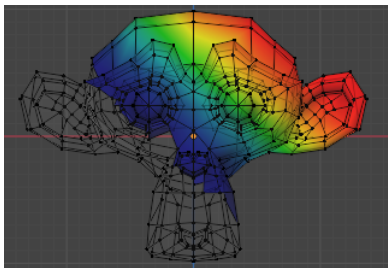
ディスプレイを追加して、参照する頂点グループの設定をします。
強さを適度に設定することで、このようにスザンヌに近づいた部分がへこむエフェクトを設定できました。



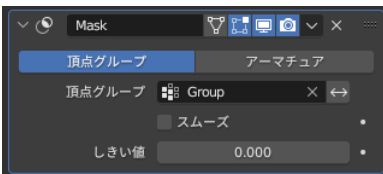
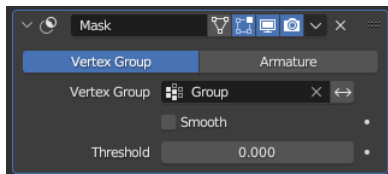
頂点ウェイト近傍のための距離の計算は、ジオメトリ設定の時はターゲットオブジェクトのスケールに依存します。
上の例でスザンヌを2倍に拡大すると、単純にスザンヌが大きくなるだけでなく、影響する範囲自体も2倍の2mになります。
影響範囲は変えずに、スザンヌを拡大縮小するときは、オブジェクトのスケールを変えるのではなく、編集モードでメッシュそのものを拡大縮小します。

マスク(Mask)


このモディファイアは、頂点ウェイトが設定されている頂点を残して残りの頂点を隠します。

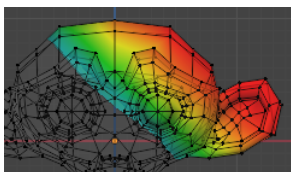


スザンヌの一部に頂点ウェイトを設定し、マスクをかけてみました。
グラデーション塗りで、斜めに頂点ウェイトを塗りましたが、
このようにウェイトが 0 の部分(青だった部分)が消えています。



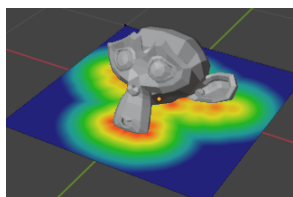
しきい値(Threshold)の設定で、ある値以上だけを残すようにできます。

 ボタンで、効果を反転させて、ある値以下だけ残すようにもできます。

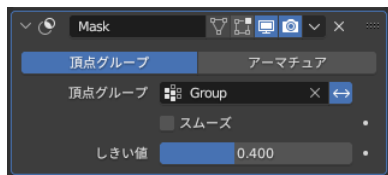


Smooth (スムーズ)を有効にすると、頂点単位での表示/非表示から、
ポリゴンの間を補間して、切り口を滑らかにすることが出来ます。

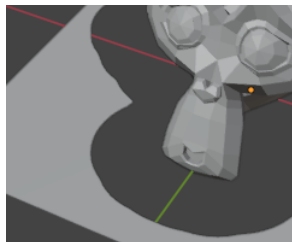
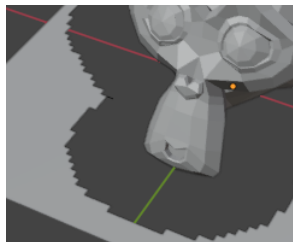
オブジェクトが接近した部分のモデルを消す



途中までは、先ほどの「オブジェクトが接近した部分をへこませる」と一緒です。



マスクを追加して、頂点グループと閾値の設定をします。
今回はウェイト値が大きい部分を残すのではなく、逆に隠したいので、反転ボタンを押しておきます。



スザンヌに近い部分が消えました。
スムーズがオフだと、グリッド形状がそのまま残ってかくかくです。
スムーズのオプションを有効にすれば、かなり滑らかにマスクされます。

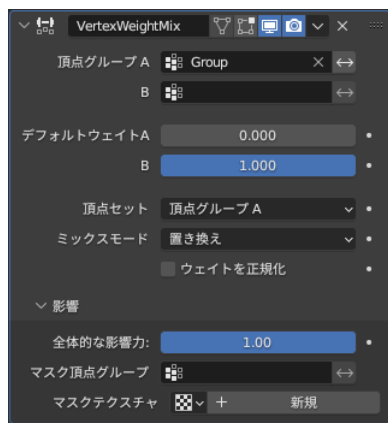
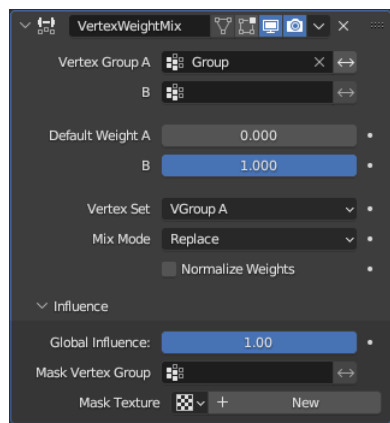
頂点ウェイト合成(Vertex Weight Mix)

このモディファイアは、2つの頂点ウェイトをその名の通り混ぜ合わせます。
足し合わせたり、かけ合わせたりといった合成処理をするのが基本的な使い方です。

ところが、この合成処理にテクスチャを適用することができるので、テクスチャの情報を他のモディファイアに伝えるという副次的な使い方をすることもできます。

ここでは、モディファイア芸での使い方としてこの使い方を見えます。

このモディファイアも少々パラメータが多くややこしいので、簡単に解説します。



頂点グループAとBを設定します。
グループAに対して、グループBの頂点ウェイトを足したり引いたりして、その結果が「**グループA**」に反映されます。

グループBが空の時には、デフォルトウェイトBの値が使われます。
「グループAに0.25を足したい」というような時には、グループBを空にしてデフォルトウェイトを0.25にします。

ミックスモード(Mix Mode)で行う計算を決めます。加算や減算、置き換えなどの一覧から指定します。

結果が反映される時の影響範囲を決めるのが、頂点セットです。

頂点セットに頂点グループBを設定して、Bのウェイトがメッシュの左半分だけに設定されているようなときには、左半分だけしか結果が反映されません。

! デフォルトでは頂点グループAかつBなので、頂点グループBが設定されていないと計算結果が反映されないで注意です。

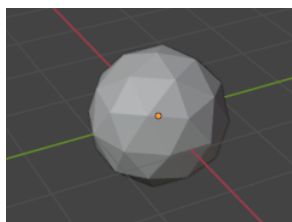
上のような設定だと、頂点グループAのウェイトが、頂点グループBのウェイトに（この場合は設定されていないので、デフォルトウェイトとして1に）置き換わります。

応用として、例えばテクスチャで設定した模様を頂点ウェイトに反映させるようなことも可能です。

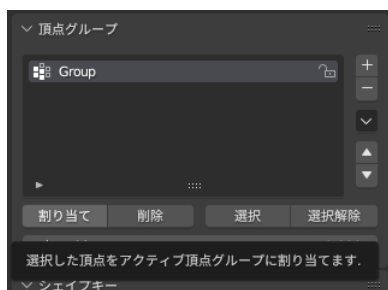
例えば上のような、ウェイト1に置き換わるような設定をしたとします。

最初のグループAの頂点ウェイトを0にしておいて、テクスチャマスクを設定すれば、テクスチャによって効果が影響されるので、結果として頂点グループAにテクスチャの値が置き換わることになります。

パターンに従って格好よくモデルを消す

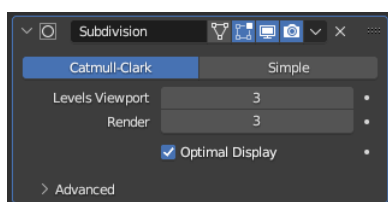


ICO球を作成します。

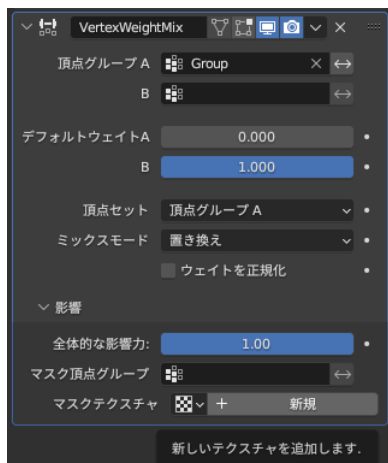


頂点グループを追加して、頂点全部にウェイトを設定します。

本当は0を設定したいところなのですが、サブディビジョンサーフェスで頂点を増やしたときにウェイト値を保持しないといけなかったので、0.01などを指定しておきます。



メッシュをサブディビジョンサーフェスで細分化しておきます。



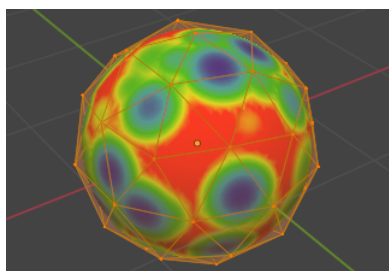
頂点ウェイト合成を追加して、上記の解説に従ってパラメータを設定します。

新規テクスチャを作成して、横のボタンからテクスチャの設定画面を表示します。

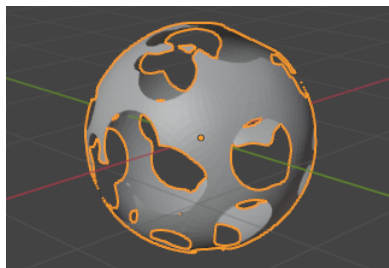


テクスチャとしてポロノイ図形を使ってみます。

座標系やスケールなどを適度に調整します。



頂点ウェイトの値を見ても、ポロノイのパターンに置き換わっていることがわかります。



後はマスクモディファイアを追加して、適切なパラメータを設定すれば、パターンに従ってモデルが消えます。

スムーズを有効にすれば、滑らかな形状にできます。