

≡ 挫折させない ≡

HoudiniTM

ドリル

（ 作例データと解説動画で
くりかえしたたきこむ ）

新井克哉 著



試し読み

for preview only

定価：3,960 円
(本体 3,600 円+税 10%)

≡ 挫折させない ≡

HoudiniTM

ドリル

（ 作例データと解説動画で
くりかえしたたきこむ ）

新井克哉 著

ダウンロードデータと書籍情報について

本書のウェブページでは、ダウンロードデータ、追加・更新情報、発売日以降に判明した誤植（正誤）などを掲載しています。

また本書に関するお問い合わせの際は、事前に下記ページをご確認ください。

<https://www.borndigital.co.jp/book/9784862466136/>

■ご注意

本書は著作権上の保護を受けています。論評目的の抜粋や引用を除いて、著作権者および出版社の承諾なしに複製することはできません。本書やその一部の複写作成は個人使用目的以外のいかなる理由であれ、著作権法違反になります。

■責任と保証の制限

本書の著者、編集者および出版社は、本書を作成するにあたり最大限の努力をしました。但し、本書の内容に関して明示、非明示に関わらず、いかなる保証も致しません。本書の内容、それによって得られた成果の利用に関して、または、その結果として生じた偶発的、間接的損傷に関して一切の責任を負いません。

■商標

本書に記載されている製品名、会社名は、それぞれ各社の商標または登録商標です。本書では、商標を所有する会社や組織の一覧を明示すること、または商標名を記載するたびに商標記号を挿入することは特別な場合を除き行っていません。本書は、商標名を編集上の目的だけで使用しています。商標所有者の利益は厳守されており、商標の権利を侵害する意図は全くありません。

はじめに

HoudiniはSide Effects Software社が開発・販売しているソフトウェアです。映像からゲーム制作まで幅広く使われており、本書を手にとった皆さんは昨今の盛り上がりをご存知かと思います。

しかしながらHoudiniは「難しいソフト」と呼ばれ、多くの挫折者を生むことでも有名です。そこで本書ではすでに挫折を経験した方々や恐る恐る扉を叩く方々に向けて、あなたの行く手を阻む難問を丁寧に1つずつ取り除き、**Houdiniを自分の力で使いこなすこと**を目標に、徹底的にサポートする内容になっています。

意味も分からず手順だけをコピーする(How)のではなく、**なぜそのノードを使うのか**という力強いロジック(Why)を身につけましょう。

もちろん座学だけでは知識の定着が難しいため、実際に手を動かしながらより良い設計ができるようステップアップしていきましょう。

昨今の技術は目を見張るスピードで進歩を遂げていますが「ただそれを使っただけ」の出力に意味はありません。大切なのは「その道具を使って自身の表現ができるようになること」です。

自分の力で、イメージ通りツールをコントロールできるようになったという実感があなたの背中を押してくれます。

学習を通して得た1つひとつの小さな成功体験が、かけがえのない財産になってくれるはずです。

さあ、一緒にHoudiniの扉を叩きましょう。

新井 克哉 a.k.a めんたいこ

CONTENTS

1 章 事前準備.....002

Houdini 無料版のインストール	002
ユーザーインターフェースと基本操作	005
コンテキストについて	008

2 章 基本操作.....012

ビューポート操作	012
Houdini の仕組み	020
アトリビュートとは：概念	020
アトリビュートとは：実践	021
アトリビュートとコンポーネント	030
エクスプレッションについて	035
グローバル変数・ローカル変数について	038
プリセットの利用	049
グループコントロール	054
コンポーネント深掘り	067
アトリビュート深掘り	071
コピーマスターへの道	100
Attribute Wrangle について	120
様々なノード	161
Attribute Wrangle・エクスプレッション脱初心者編	217
ベクトルの演算	218

VEX 脱初心者編	222
アトリビュート操作復習	222
point 関数とベクトル計算	226
point 関数とブレンドシェイプ	232
fit 関数	236
プリミティブのランダム削除	246
Attribute VOP について	248

3 章 考える（考え続ける）訓練編 251

カプセルのモデリングをしてみよう	251
積雪ツールを作ってみよう	263
建築に応用できる柱の配置を行ってみよう	269
滑らかな曲線でメッシュをカットしてみよう	280
モーフテキストアニメーションを作ってみよう	287
任意の点からラインを引いてみよう	294
旗を作ってみよう	301
隣接する円を作ってみよう	312
大小様々な形の岩を作ってみよう	318
コピーをコントロールしよう	327
まとめ	335

1 章 ▶ 事前準備

Houdini の世界に飛び込む前に、しっかりと事前準備をしておきましょう。

≫ Houdini 無料版のインストール

本書は Houdini FX、CORE、Indie、Education および Apprentice (無料版) すべてのバージョンで読み進めることが可能です。以下では Houdini をまだ入手していない方のために、無料版の導入方法についてご説明します。

▶ 無料版について

無料版ではいくつかの制限を除き、有料版のほぼすべての機能を使うことができます。学習目的で無料版を使用した後、有料版にステップアップする際にはご自身の目的に合わせてライセンスを選びましょう。

ライセンスについての詳しい情報は「SideFX 社ホームページ: <https://www.sidefx.com/ja/>」をご覧ください。

▶ 無料版のダウンロード

まずは SideFX のアカウントを作成し、「ダウンロードページ: <https://www.sidefx.com/download/>」からインストーラをダウンロードします。

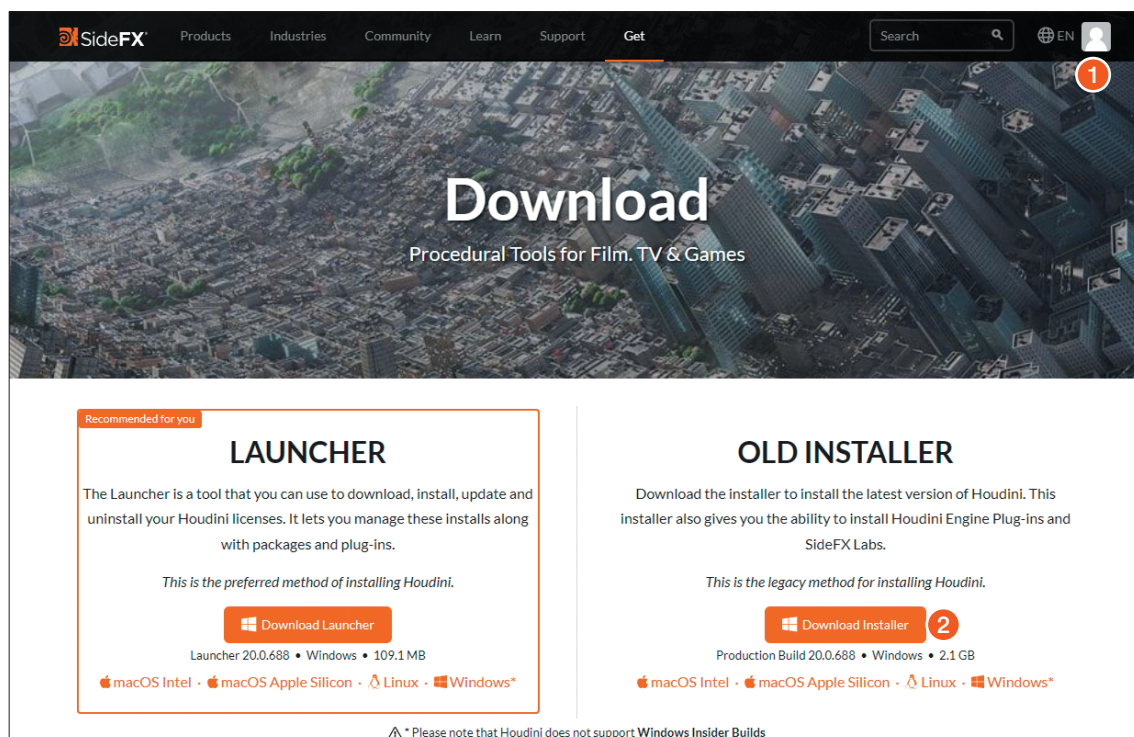


図 01-001 インストーラのダウンロード

① ログインすると Login の表記からアイコンの表記に変わります。

- ② インストーラはバージョンの管理ができる LAUNCHER とバージョン固定の OLD INSTALLER を選べますが、筆者は問題が発生したときにバージョンの切り分けを行いやすいため、OLD INSTALLER を使用しています。そこで本書では、OLD INSTALLER をダウンロードして説明を行います。

▶ 無料版のインストール

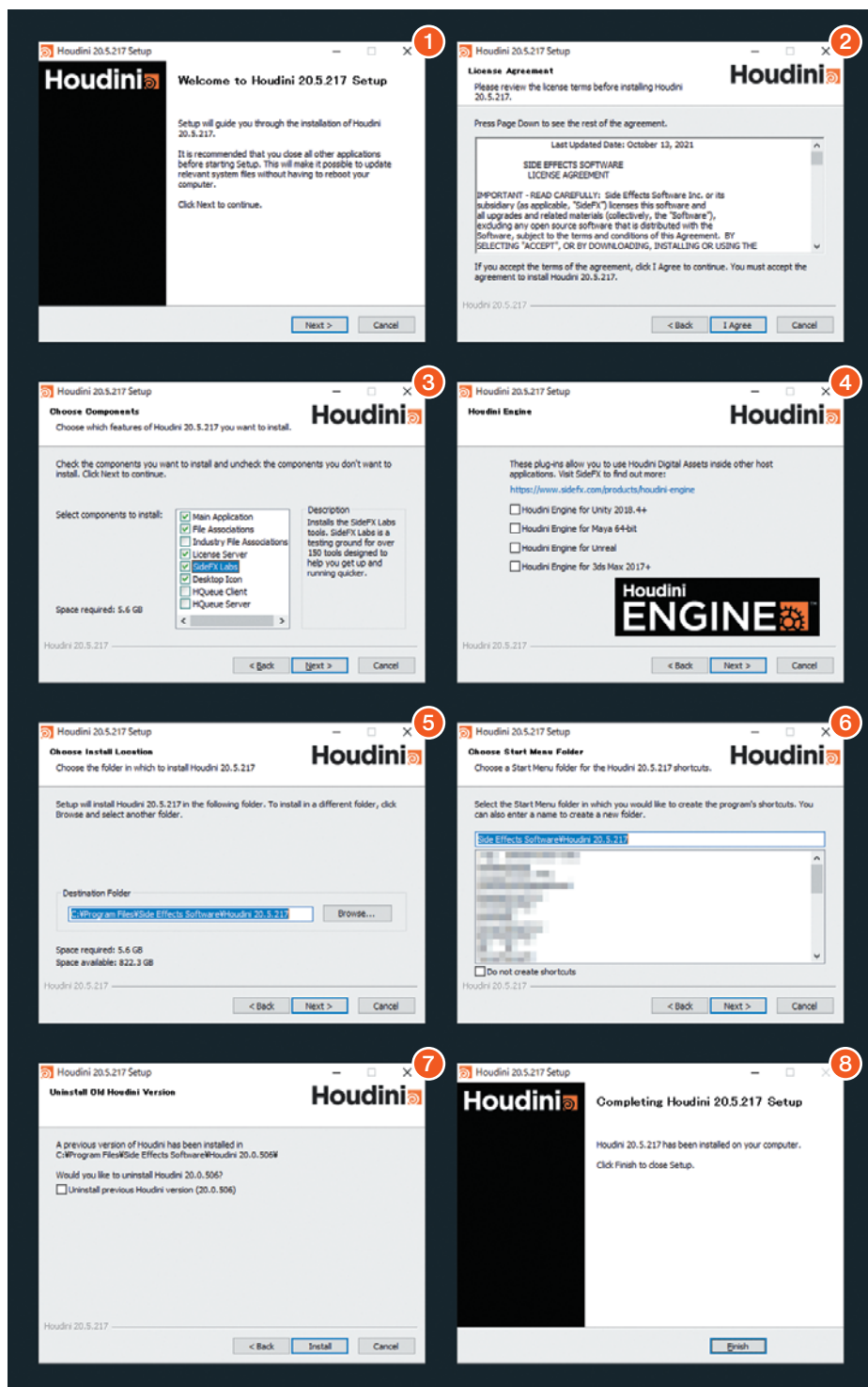


図 01-002 インストール手順

ダウンロードしたインストーラを起動します。ライセンス要項を確認してインストールをしていきます。
ここでは特筆事項のみ解説していきます。

- ③ 本体インストール時に追加のコンポーネント（便利なツールなど）もインストール可能です。今後役に立つケースが多いため、SideFX Labsにチェックを入れることをおすすめします。
- ④ Houdini を Maya、3ds Max、Unity、Unreal Engine と連携するための Houdini Engine というプラグインをインストールすることができますが、本書では割愛します。
- ⑦ 過去のバージョンをインストールしている場合は古いバージョンを削除するかどうか決定します。業務で Houdini を利用するようになったら安全のため過去のバージョンを保持しつつバージョンアップの検証を行うとよいでしょう。

▶ 無料版の起動

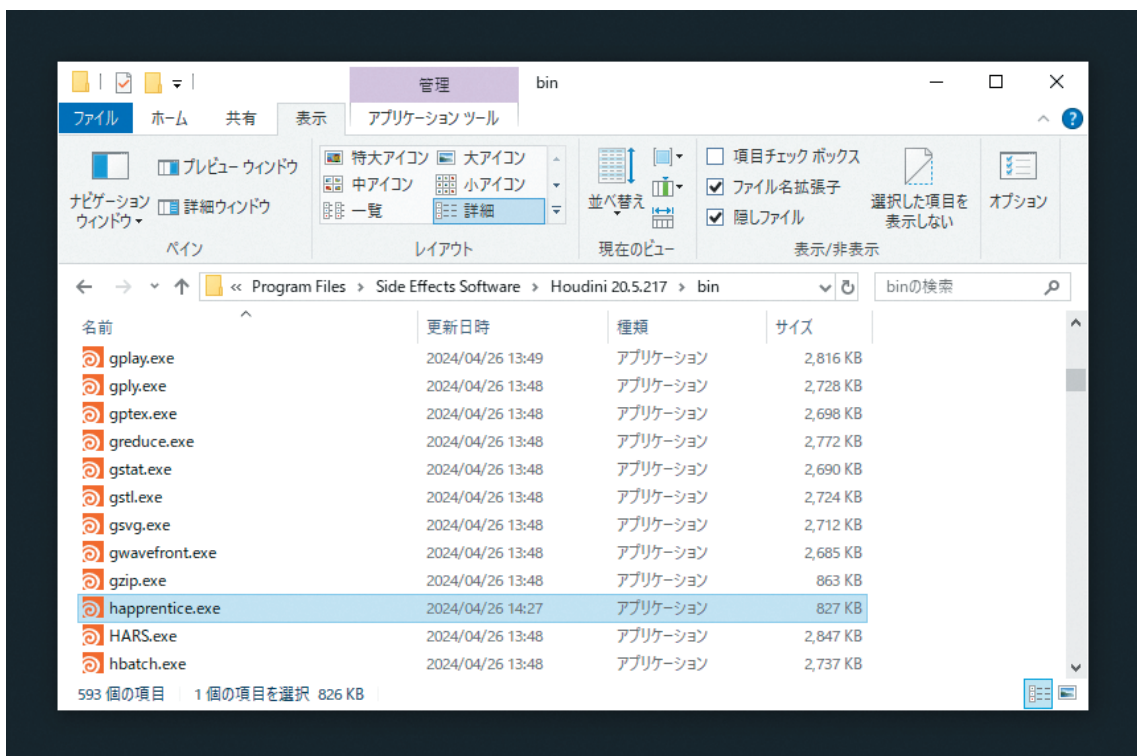


図 01-003 Houdini のアプリケーションが置かれたフォルダ

無料版の Houdini は Houdini Apprentice と呼ばれ、Windows 環境でデフォルト設定でインストールした場合は C:/Program Files/Side Effects Software/Houdini のバージョン /bin 以下にアプリケーション本体が保存されています。

happrentice.exe を起動して Houdini ライフを始めていきましょう。

≫ ユーザーインターフェースと基本操作

Houdiniではユーザーインターフェース（以降UIと略します）を自分好みに変更し、Desktopという単位で管理することができます。

本書はデフォルトのUIのままでも読み進めることができますが、手元と比べて大きく画面が違くと学習しにくい可能性もあるので、筆者が業務で使用しているDesktopをご紹介しますながらDesktopの編集・保存方法を見ていきましょう。

その上でHoudiniの基本操作をご紹介します。

▶ Desktopの変更・保存

画面上部のDesktopタブを見ると、デフォルトでは「Build」となっているのが確認できます。このデスクトップは他のDCCツール*1と同様、シーンビュー（ビューポート）が大きく表示されているものになります。

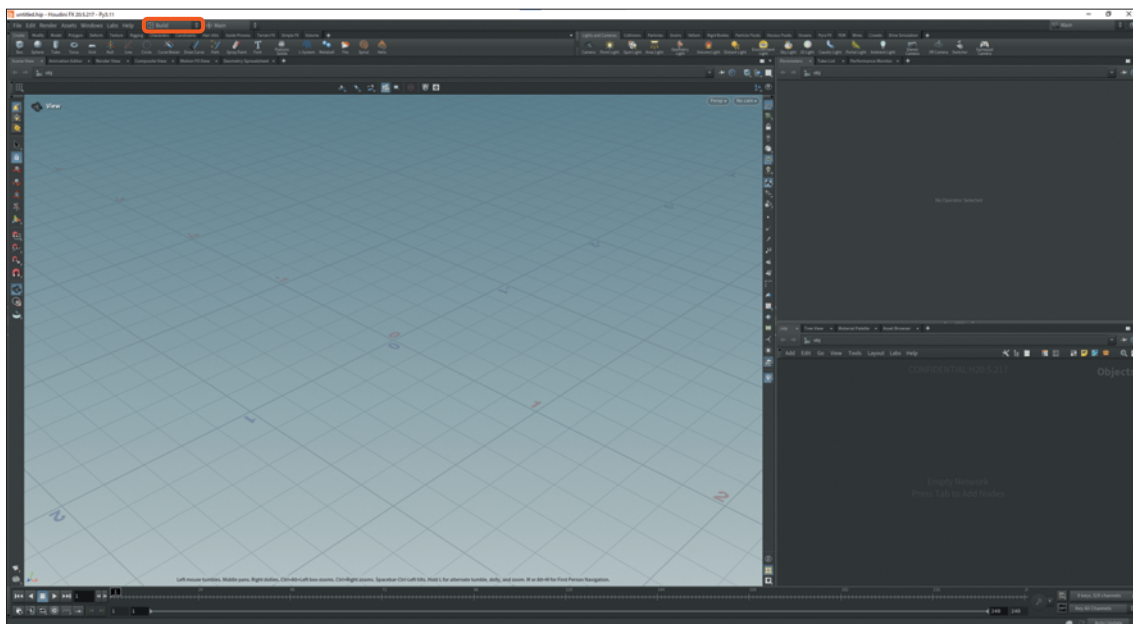


図01-004 デフォルト起動画面：Buildデスクトップ

他のツールから移行してきた方には馴染みが深いUIだと思いますが、Houdiniでは後述するように次のペイン（ウィンドウ）がとても重要になってきます。

- ・ネットワークビュー（Network View）
- ・ジオメトリスプレッドシート（Geometry Spreadsheet）

ネットワークビューはノードを配置したり繋いだりする場所です。ジオメトリスプレッドシートは表計算ソフトのようにジオメトリの状態を一覧で確認できる場所*2です。

重要なところを大きく表示したほうが作業が捗るので、既存のデスクトップを改造していきましょう。「Build」

*1 DCCとはDigital Content Creationの略で、3D統合開発ツールのことを指します。簡単に言うとMaya、3ds Max、Blenderなどのソフトのことです。

*2 この説明ではわかりにくいと思いますが、後ほど丁寧に解説しますので安心してください。

を改造しても良いのですが、目的に近い「Technical」からスタートします。

Desktop タブをクリックし複数のデスクトップ候補から、ここでは Technical を選択します。

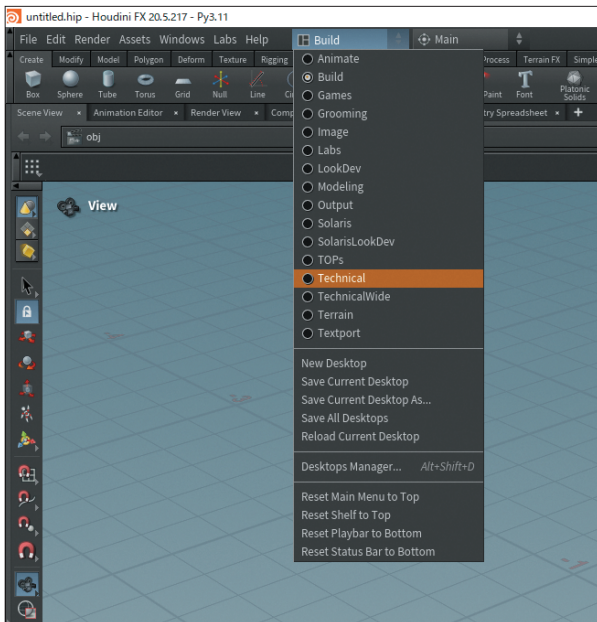


図 01-005 デスクトップ選択メニュー

こちらが Technical デスクトップです。ここから改造していくのですが、スクリーンショットより動画をご覧くださいほうがわかりやすいと思うのでご参考ください。

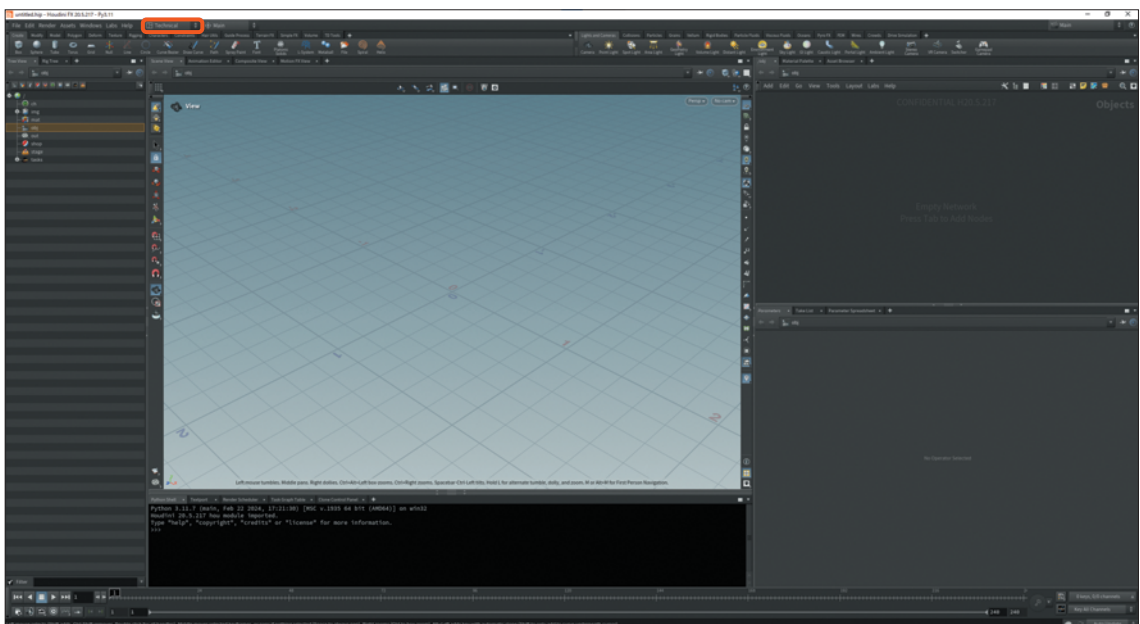


図 01-006 Technical デスクトップ



デスクトップのカスタマイズ

[edit_desktop.mp4](#)

デフォルトの Technical デスクトップをカスタムしているところ

変更し終わったらデスクトップを保存します。「Save Current Desktop As…」を実行して別名保存しましょう。

今回は「StartKit」とします。

これでデスクトップが保存できました。本書と同じUIで進めたい場合はこちらのデスクトップをご利用ください。

▶ インターフェースの概要

Desktopを使いやすく変更したので画面の名称などを押さえておきましょう。本書を読み進めていく上でも、インターネットなどの情報を探す上でも基本となる部分です。

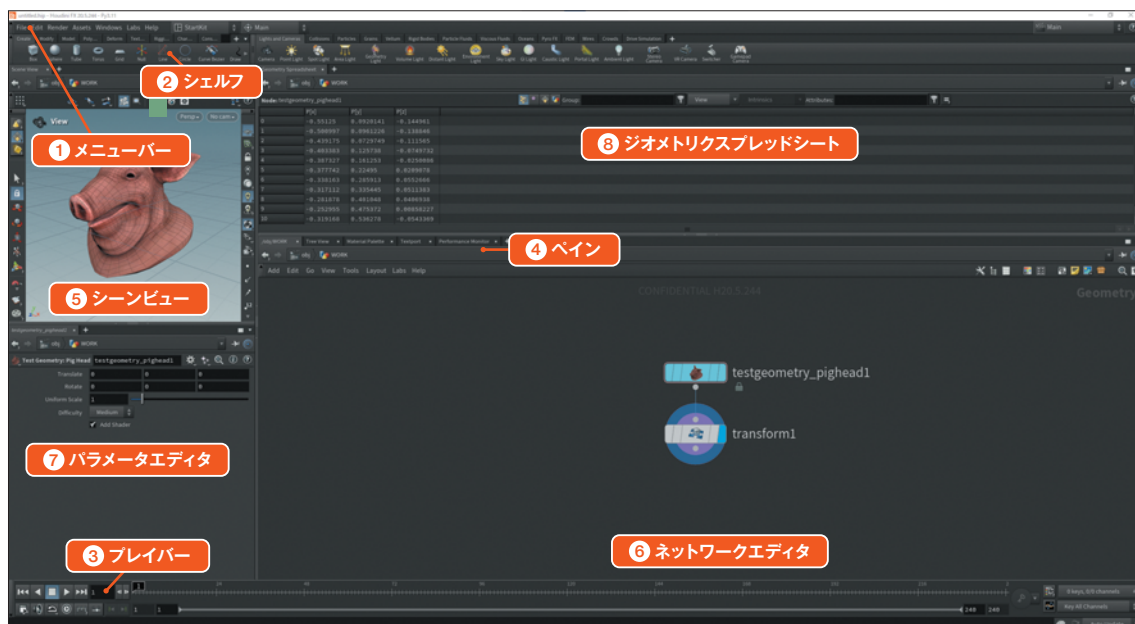


図01-007 インターフェース

▶ ①メニューバー

メインメニューとも呼ばれます。ファイルのインポートやレンダリングの管理など、よく使う機能にアクセスすることができます。多くのDCCツールも同様なUIを持っているのでわかりやすいでしょう。

▶ ②シェルフ

海や爆発、破壊などをワンボタンで作ってくれる便利機能がまとまっています。ですが筆者はデフォルトで用意されたシェルフはほぼ業務で使用しません。その理由としてネットワーク(ノードの組み方)の構成が筆者の好みと異なること、および大量のノードが一気に生成されてしまうため管理がしにくいことなどが挙げられます。

▶ ③プレイバー

アニメーションを再生するためのフレームを制御します。

▶ ④ペイン

ペイン (Pane) は英語で窓枠という意味で、複数のウィンドウをタブで管理することができる領域です。

▶ ⑤シーンビュー

3Dオブジェクトを表示する領域です。Desktopの編集のときにお話しましたが、個人的にはHoudiniにおいてメインの領域ではなく、デバッグ画面としてとらえておくことをおすすめします。

▶ ⑥ ネットワークエディタ

ノードを操作するHoudiniの最重要ウィンドウです。Houdiniでは組み上げたノード全体をネットワークと呼びますので覚えておく和良好的でしょう。

▶ ⑦ パラメータエディタ

ノード固有のパラメータを操作するウィンドウです。こちらも重要なウィンドウなので、ホットキーで表示・非表示する機能もあります。そこについては基本操作編で紹介しましょう。

▶ ⑧ ジオメトリスプレッドシート

ジオメトリの状態を確認することができるウィンドウです。初心者の方は数字が大量に並んでいてひるんでしまうかもしれませんが、**Houdiniの最重要機能の1つ**です。この機能のおかげでデバッグが大いに捗り、堅牢なシステムを作ることが可能になります。慣れてくるとありがたみが身にしみる、素晴らしい機能です。

コンテキストについて

実際にHoudiniを触り始める前に、**コンテキスト**という考え方だけは先におさえておきましょう。この知識があるのとないのでは今後の学習に雲泥の差が生まれます。

はやる気持ちをグッと抑えて、まずは座学にお付き合いください。そしてここでは理解を深めるためにBlenderの画面を参考に説明します。

ここで使用しているBlenderはバージョン3.6.5 LTSとなっております。本書はHoudiniの解説書であり、Blenderの説明も**一般的なDCCツール**の代表として登場してもらっているだけなのでバージョンの差異は無視していただいて構いません。また、Blenderを使用したことがない方は無理にインストールする必要もホットキーを覚える必要もありません。多くのツールで用いられている概念が伝わればよいので、ご自身の馴染みが深いツールに置き換えてお読みください^{*1}。

■ Blender

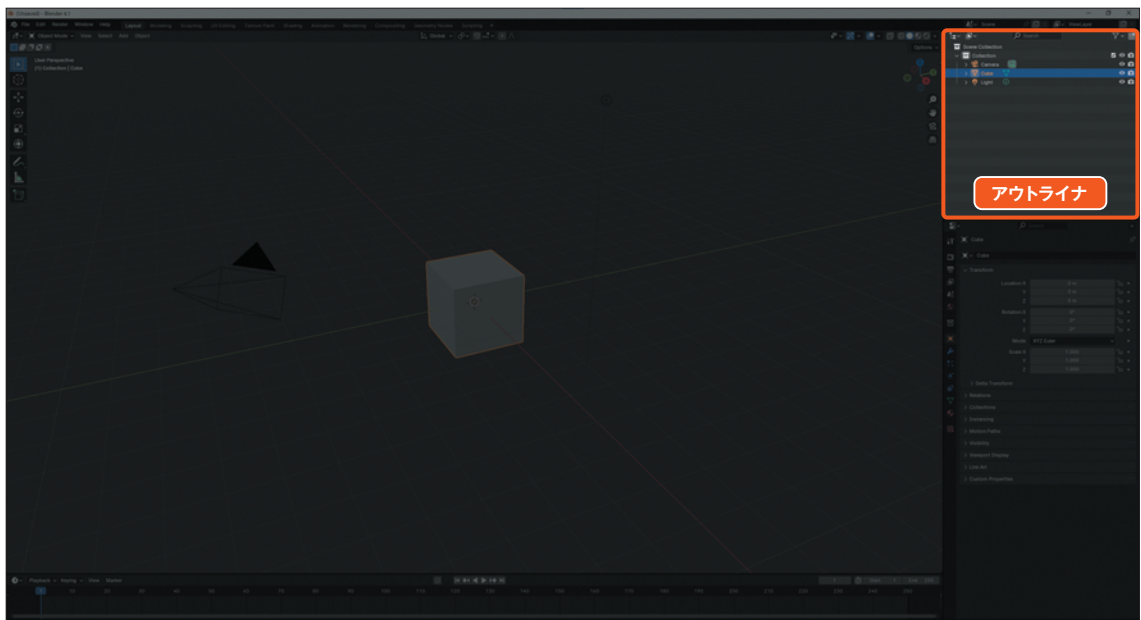


図 01-008 Blender の UI

ここでお話をする重要な概念は次の2点のみです。

- ・アウトライナ
- ・エディットモード

アウトライナは、このシーンにどんなオブジェクトが置いてあるのか、それが目に見えるのかどうかなどのシーン全体の状態を管理する場所です。

多くのDCCツールにもゲームエンジンにも存在する機能なのでご存じの方も多いでしょう。続いてHoudiniの説明をしますが、その際このアウトライナに該当する場所がどこにあたるのか意識しておきましょう。

続いてエディットモードに関してですが、選択されたオブジェクトのポリゴンモデリングを行うモードのことです(Blenderの場合は[TAB]キーでエディットモードに入れますね)。

オブジェクトそのものの操作を行いたい場合はもう一度[TAB]キーを押してオブジェクトモードに戻ります。

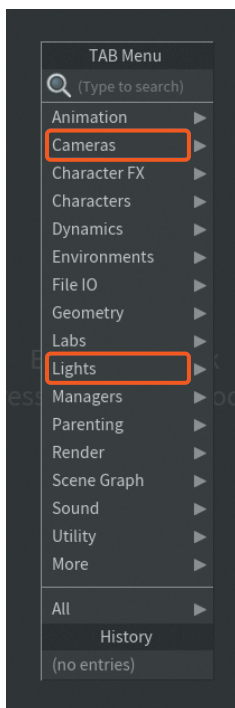
オブジェクトモードとエディットモードを行き来してモデリングを行っていく、ということを覚えておきましょう。

■ Houdini

それではHoudiniに戻ってきましょう。操作方法については後ほど詳しく解説しますので、まずは本書と同じ操作を試してみてください。

Houdiniを起動したら、マウスをネットワークエディタ上に移動させて[TAB]キーを押します

*1 Houdiniが初めて触るDCCツールだという方もおられると思いますが、個人的には1ヶ月でもいいのでBlenderやMaya、3dsMaxなどの一般的なDCCツールを勉強されることをおすすめします。Houdiniは非常に自由度が高いツールなのですが、その「自由さ」を理解するためには「普通の」考え方を先に学ぶ必要があると筆者は考えています。

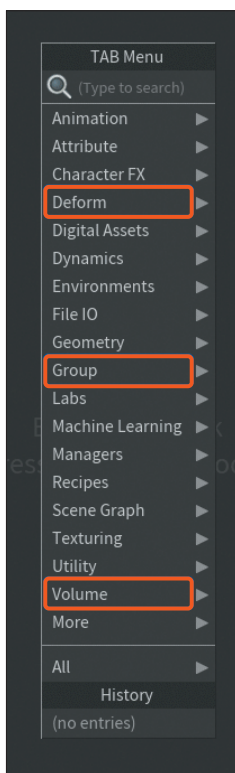


左の図のような状態になりましたか？ 簡単に操作の説明をすると[TAB]キーを押すことで今作ることができるノードを一覧表示してくれるという仕組みになっています。この「今作ることができる」ということについて丁寧に説明していきますが、まずはこの一覧の並びをなんとなく覚えておいてください。後に見比べるので厳密でなくてOKです。

上記の操作に続けて以下を行ってください。

- 1 (Tabメニューが出ている状態で) geo と入力してください
- 2 Geometry という候補がハイライトされたら [Enter] キーを押します
- 3 ノードを作りたい場所にマウスを動かし、[Enter] キーを押すと Geometry ノードが作られます

図 01-009 ネットワーク
エディタ上、OBJレベルで
[TAB]キーを押す



初めてノードを作ってみましたね。ここまでは分かるのではないのでしょうか。しかしここからが本題なので気を引き締めて行きましょう。

- 1 Geometry ノードをマウスで選択した状態で [Enter] キーを押します (マウスのダブルクリックでも OK です)
- 2 Geometry ノードの中に入ることができるので、そこで [TAB] キーを押します

図 01-010 ネットワーク
エディタ上、SOPレベルで
[TAB]キーを押す

先程と同様、今作ることができるノードが一覧表示されました。しかし、先程の図 01-009 とは様子が変わり、ノード一覧の内容が異なります。

Geometry ノードを作ったときには候補に Cameras や Lights がありましたが、Geometry ノードの内部では Deform や Group、Volume などが表示されています。

Houdiniはすべてがノードでできており、ノード1つひとつは[TAB]キーを押すことで呼び出すことができます。その総量は膨大で、もし[TAB]キーを押した瞬間全部のノードが候補で出てきてしまったら目的のノードを探すのも一苦勞になってしまいますよね。

それを防ぐためにSideFXは「マテリアルを作りたいときはマテリアルに関係あるノード」だけ、「モデリングをしたいときはモデリングに関係あるノード」だけ、「コンポジットをしたいときはコンポジットに関係あるノードだけ」をあらかじめ絞って表示してくれる賢い機能があります。これがコンテキストと呼ばれるものです。

この機能を念頭に置いて先程の作業を振り返ってみましょう。まず最初にgeoと入力してGeometryノードを作成しました。これはシーンにオブジェクトを配置することを意味します(シーンに配置するものが候補で出てくるので、CameraやLightが表示されていたわけですね)。

ここがBlender(または他のDCCツール)で言うところのオブジェクトモードです。つまりアウトライナと同じシーン全体の状態を管理する場所ということです。

そして、Geometryを選択して[Enter]キーを押す、またはマウスのダブルクリックで中に入ったあとではノードの候補はモデリングに特化したもの*1に絞られます。この場所がBlender(他のDCCツール)で言うところのエディットモードとなります。

これらをまとめると次の通りです。

- 1 Houdiniは「今やりたい作業」に合わせてノードの候補が変わる仕組みがあり、これをコンテキストと呼びます。
- 2 カメラやライト、ジオメトリを作る場所はObjectやOBJ、Sceneレベルと呼ばれます。これは一番上の階層で、シーン全体を管理する場所です。
- 3 個別のジオメトリノードの中では主にモデリングを行います。Surfaceを編集する機能を持つことからSurface OPerator、略してSOP(ソップ)と呼ばれます。

Houdiniではノードのことを○○ノードと呼んでも良いですし、○○「コンテキスト名」と呼ぶこともあります。例えばモデリングをする場所であるSOPで使うPolyBevelはPolyBevelノードと呼んでもよいですし、PolyBevelソップと呼んでもOKです。

本書ではモデリングをする場所であるSOPを徹底的に学んでいきます。「本当はエフェクトをやりたいんだけど」や「プロシージャルモデリングには興味がないな」という方も、どうかここから学習をスタートさせてください。この先すべてのHoudiniの作業には、このSOPの理解が密接に関わってきます。

「SOPを制するものはHoudiniを制す」という言葉もあります。一緒に学んでいきましょう。

Column

Windowsを使っている方はコンテキストメニュー(右クリックメニュー)はご存知かと思います。これはフォルダ上で右クリックとファイル上で右クリックで表示されるメニューが異なる機能ですが、まさにこれがHoudiniのコンテキストと同じものです。コンテキストとは日本語でいうと「文脈」に当たるので、現在の文脈に合わせて振る舞いを変えてくれる気の利いた機能というわけですね。

*1 実際はモデリングのみではないのですが、まずは大枠でモデリングととらえた方が理解が深まるでしょう。

2 章 ▶ 基本操作

[TAB]キーで「今やりたい作業」に合わせてノードの候補を出す、というのは前の章で説明しました。これが最も重要なHoudiniの操作なのですが、それ以外も大切な操作が多くありますので、簡単にご紹介しておきましょう。

ポイントとして、**Houdiniはマウスがどのウィンドウ上にいるかで操作が変わる**ということを覚えておくとい良いでしょう*1。

画面操作に関しては動画の方がわかりやすいところもあると思いますので、動画をご用意しました。

≫ ビューポート操作

ビューポート（シーンビューとも呼びます）の操作はDCCツールの基本操作です。

ビューポートでは大別するとポリゴンやポイント、エッジなどの選択を行うセレクトモードとカメラ操作を行うビューモードがあり、ビューポート上にマウスがある状態でホットキーを押すことでモードを切り替えることができます。

ホットキー	モード
	セレクトモード
	ビューモード

キーを押すごとにビューポート横のセレクトボタンとビューボタンがそれぞれアクティブに切り替わります。もちろんこのやり方で操作しても良いのですが、**一時的にビューモードに切り替えたい**というケースが多々あります。その際は[SPACE]キーを押しながらマウス操作を組み合わせる方法を使います。組み合わせ方法は次の表のとおりです。

オペレーション	操作内容
 + 左ドラッグ	カメラの回転（タンブル）
 + 右ドラッグ、またはホイール回転	カメラの前後移動（ドリー）
 + 中ドラッグ	カメラの平行移動（トラック）
 + 	カメラがホームポジションに移動

また[SPACE]キーの代わりに[Alt]キーでも同様の操作を行うことができます。好みのキーを使用しても良いですが、何も考えずいろいろな方向からオブジェクトを確認できるよう、手に馴染むまで訓練してください。



ビューポートでのカメラ操作

[viewcamera.mp4](#)

カメラ操作をしっかりと覚えましょう

*1 意図した結果にならない場合はマウスが重なっているウィンドウを確認するようにしましょう。

またマウスカースルをシーンビュー上においた状態で下記ホットキーを押すことでビューを切り替えることができます。Houdiniの操作に慣れてきたら積極的に使用していきましょう。

ホットキー	機能
SPACE + 1	パースペクティブビュー
SPACE + 2	トップビュー (2回押すとボトムビュー)
SPACE + 3	フロントビュー (2回押すとバックビュー)
SPACE + 4	ライトビュー (2回押すとレフトビュー)
SPACE + 5	UVビュー

実際の動作は次の動画をご参考ください。



ビュー操作のホットキー

[viewport.mp4](#)

シーンビューは、見え方をホットキーで切り替えられます

▶ ネットワークビュー操作と処理の流れ

Houdiniはその時にやりたいこと、つまりコンテキストによって作成するノードの候補を自動的に制限してくれるというお話はすでにしましたね。

そしてこのコンテキストによって挙動が変化するのはノード自体も当てはまります。ここでは本書で多く扱うOBJレベルとSOPレベルのノードを見比べてみましょう。



図02-001 Geometry ノード

上の図はOBJレベルに配置したGeometryノードです。このノードにはディスプレイフラグとセレクトابلフラグがあります(フラグはボタンのようにクリックするとオン・オフできます)。

ディスプレイフラグはビューポート上での表示・非表示を示し、セレクトابلフラグはビューポート上で選択可・選択不可を切り替えます。

OBJレベルではシーンを構築し、複数のものを配置するため、各々のノードにはビューポート上での見え方や入り組んだ形状を操作するときに誤って選択してしまうのを防いだりするフラグがあるわけですね。

以下にSOPレベルのノードもご紹介しますが、フラグの数が変わり、かつ同じ「ディスプレイフラグ」という名前の機能も意味が変わってきます。見ていきましょう。

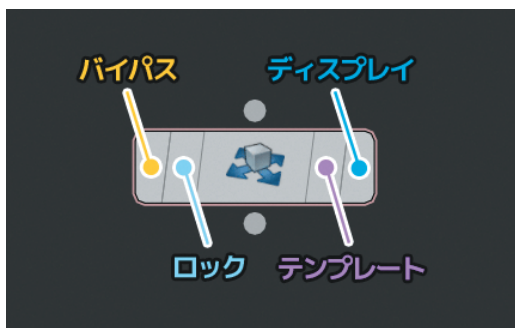


図 02-002 Transform ノード

上の図は SOP レベルに配置した Transform ノードです。フラグが4つになりましたね。これらフラグはストリーム（ノードの流れ）を制御する上でとても大切なのでここでしっかり学んでいきましょう。

■ディスプレイフラグ

SOP レベルのネットワークで最も重要なのがこのディスプレイフラグです。先述のとおり、OBJ レベルのノードにもディスプレイフラグはありますが、その意味はまったく異なります。

SOP ではディスプレイフラグがオンになっているノードまでネットワークを計算します。

つまりディスプレイフラグが立っているノード以下にノードをたくさん繋いでいても無視され、ディスプレイフラグが立っているノードまでの計算結果がビューポートに表示されることになります。

初学者のうちはこの仕組みを不思議に思うかもしれませんが、ディスプレイフラグを切り替えるだけでどこまで計算をするか（つまりどのネットワークまで有効にするか）を変更することができ、複雑なネットワークを組むようになってからとても助かる機能になります。

つまりディスプレイフラグは「どこまで計算をするかを示すゴール」と言えます。そのため、**1つのネットワークでディスプレイフラグをオンにできるのは1つのノードだけ**となります。ゴールは1つだけ、と覚えておきましょう^{*1}。

また本書を読み進めていく中で「ノードを繋いでも結果が変わらない」という場合は、このディスプレイフラグがそのノードでオンになっているかを確認するようにしましょう。

■バイパスフラグ

ノードの機能を一時的にバイパスする（無効化する）フラグです。このフラグはネットワーク内で複数のノードでオンにすることができ、ノードのオン・オフによって結果がどのように変わったか確認する際に非常に有用な機能です。

■テンプレートフラグ

このフラグが立っているノードはワイヤーフレームで表示されます。複数のノードを左マウスドラッグで選択、もしくは[Shift]キーを押しながら複数のノードを選択した後フラグをクリックすることで複数のノードをテンプレート表示することができます。

^{*1} ネットワーク内のノードを参照して、計算することをクッキングと言います。なぜクッキングフラグではなくディスプレイフラグなのでしょう。それはこのフラグがあくまで3Dビューポートに表示（ディスプレイ）させる場所を指定しているのであり、ビューポートに表示させる過程で結果的に計算処理が走っているためです。ビューポートは1つしかないのにディスプレイフラグも1箇所しか指定できませんね。ディスプレイには別のものを表示させつつ、Render結果は別のレンダーフラグとして指定することも可能です。

テンプレート表示した状態とディスプレイフラグが立っている状態を見比べる際に利用します。

■ ロックフラグ

このフラグが立っているノードまでの計算結果をシーンに保存します。ロックフラグを立てた後は上流の計算結果が固定されるので、上流のネットワークをまるっと削除しても計算結果は保持されます。

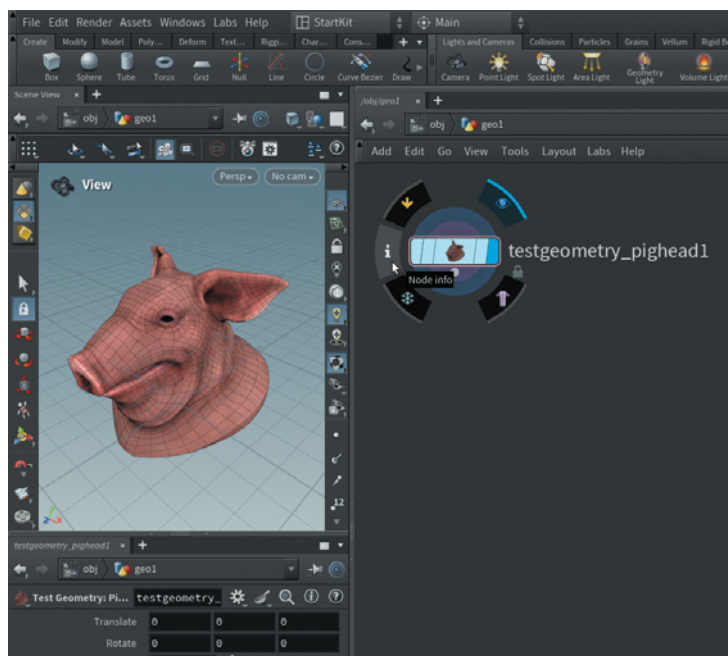
一見便利そうに聞こえる機能ですが、筆者はほとんどこの機能を利用しません。なぜならば、上流の変更を受けて計算結果が変更されるプロシージャルワークフローのメリットが完全になくなってしまうためです。

またシーンのファイルサイズが肥大化するという弊害もあります。Houdiniは基本的に各ノードのパラメータのみをシーンに保存するので、非常にファイルサイズが小さくなるという特徴がありますが、このロック機能を使うとその計算結果がシーンに丸ごと保存されてしまうためです。

よって本書でロックフラグを利用することはありませんし、実務上も特殊なケースでなければ利用しないというスタンスをおすすめします。

▶ ノードリングとフラグ、そして Node Info

ノードの上にマウスオーバーするとノードリングというものが表示されます。これは上述のフラグ操作と同様の機能を持ったUIなのですが、ネットワークが広くなってきたときにノードを小さく表示していてもフラグ操作をしやすいよう追加された機能です。しかし、ノードリングにはノードのフラグにはない機能が1つあり、それがNode Info（ノードインフォ）と呼ばれるものです。



ノードリング

ノードリング左側のiマークがNode Infoになります。これをクリックするとそのノードにおけるジオメトリの情報を詳しく表示してくれます。