

---

# **nanoem Documentation**

リリース ***v34.0.0***

**2022 年 01 月 26 日**



# 目次

第 1 章	インストール	3
1.1	必要な動作環境	3
1.2	利用条件	4
1.3	ダウンロード方法	5
1.4	インストール後について	5
第 2 章	逆引き利用ガイド	9
2.1	とりあえず何をすればいいの？	9
2.2	モデルまたはアクセサリを読み込ませたい	11
2.3	モーションまたはポーズを読み込ませたい	11
2.4	モデルを動かしたい	13
2.5	モーションのキーフレームをコピー＆ペーストしたい	16
2.6	音源を読み込ませたい	16
2.7	背景動画を読み込ませたい	17
2.8	再生したい	18
2.9	エフェクトを読み込ませたい	18
2.10	アクセサリまたはモデルを特定のモデルに持たせたい（外部親）	18
2.11	物理演算の設定を変えたい	19
2.12	IK/物理演算を無効化したい	19
2.13	準標準ボーンを入れたい	20
2.14	モーショントレースしたい	21
2.15	その他よく聞かれる質問について	21
第 3 章	アプリケーションの画面概要	23
3.1	タイムライン	24
3.2	ビューポート	29
3.3	補完曲線パネル	32
3.4	カメラパネル	35
3.5	照明パネル	36
3.6	モデルパネル	38
3.7	ボーンパネル	40
3.8	モーフパネル	46
3.9	アクセサリパネル	47

3.10	ビューパネル . . . . .	50
3.11	再生パネル . . . . .	51
3.12	その他 . . . . .	53
3.13	画面変遷の履歴 . . . . .	53
第 4 章	アプリケーションのメニュー . . . . .	57
4.1	ファイル . . . . .	57
4.2	編集 . . . . .	73
4.3	プロジェクト . . . . .	80
4.4	カメラ . . . . .	88
4.5	光源 . . . . .	89
4.6	モデル . . . . .	91
4.7	アクセサリ . . . . .	94
第 5 章	アプリケーションの設定 . . . . .	97
5.1	全体設定 . . . . .	97
5.2	プロジェクト . . . . .	102
第 6 章	エフェクトの設定 . . . . .	105
6.1	オフスクリーン . . . . .	106
6.2	モデル . . . . .	107
6.3	パラメータ . . . . .	108
第 7 章	プラグイン . . . . .	111
7.1	導入方法 (macOS) . . . . .	111
7.2	準標準ボーンプラグイン . . . . .	111
第 8 章	よくある質問と回答 . . . . .	121
8.1	MikuMikuDance の違いはなんですか? . . . . .	121
8.2	モデル編集できますか? . . . . .	122
8.3	動画編集できますか? . . . . .	123
8.4	講座動画はありますか? . . . . .	124
8.5	MMD のプロジェクトファイルの読み込みはできますか? . . . . .	124
8.6	VRoid Studio で作った VRM を読み込むことができますか? . . . . .	124
8.7	縦長画像あるいは動画を出力したい . . . . .	124
8.8	透過つき動画を出力したい . . . . .	125
8.9	モーション調整のため最初の部分に無音を挿入したい . . . . .	125
8.10	nanoem をスリープさせるには? . . . . .	126
8.11	MMDAI2/VPVM とはどう違うのですか? . . . . .	126
8.12	あるモデルにモーションをふたつ以上適用したい . . . . .	127
8.13	古いバージョンがほしい . . . . .	127
8.14	アンインストールしたい . . . . .	127



8.15	よろずの質問	128
第 9 章	よくある質問と回答 (エフェクト編)	131
9.1	エフェクトの読み込みはできますか？	131
9.2	エフェクト詰め合わせが見当たらないのですが...	131
9.3	ray-mmd は使えますか？	132
9.4	ikPolishShader は使えますか？	133
9.5	MotionBlur 系が動かないのですが...	133
9.6	'#' : invalid directive が出る	134
9.7	画面が固まったかのような表示になる	134
9.8	nanoem 上で動いてると MME からどう判断すればよいですか？	135
第 10 章	トラブルシューティング	137
10.1	nanoem が開かない、起動しない	137
10.2	音源が読み込めないあるいは音がでない	137
10.3	謎の爆音が発生することがある	138
10.4	背景動画が読み込めない	138
10.5	プロジェクトが読めなくなってしまったのですが	138
10.6	プロジェクトが保存出来ないのですが	139
10.7	モデルが正しく表示されない	140
10.8	モデルを読み込んだら落ちる	140
10.9	モデルの足の部分がガクガクする	141
10.10	アクセサリを読み込んでも何も表示されない	141
10.11	アクセサリまたはエフェクトを読み込んだら落ちる	141
10.12	新規プロジェクト作成のたびにエラーが表示される	142
10.13	動画出力を行うと落ちる	142
10.14	動画出力で書き出すと意図しない (例えば真っ黒) 動画が出力される	142
10.15	動画出力で書き出すと再生では起こらなかったフレーム欠けが起こる	142
10.16	ウィンドウ画面が水色に表示される	143
10.17	カメラモーションを読み込んだら角度の X 軸が反転する	143
10.18	物理演算で貫通を起こす	143
10.19	カメラを動かすとセルフシャドウが消えることがある	143
10.20	zip を解凍したらファイル名が文字化けしたのですが	144
10.21	タイムラインとビューポートの分離はどうやったらできるの？	144
10.22	外部親設定したら意図しないモデルに紐付いた	144
10.23	一時停止を繰り返すと再生時にかくつくんですが...	144
10.24	一部文字が「？」と出て表示されない	145
10.25	VRM から変換したモデルを読み込ませるとボーン表示のみになる	145
10.26	macOS 10.15 以降で「キーボード入力監視」の確認を求めるダイアログが表示される	145
10.27	OSStatus returns 560558962 が出る	145
10.28	OSStatus returns 2003332927 が出る	146

10.29 This effect cannot be compiled due to the renderer is not OpenGL が出る . . . . .	146
<b>第 11 章 プライバシーポリシー</b>	<b>147</b>
11.1 強制終了時の情報送信 . . . . .	147
11.2 利用状況の情報送信 . . . . .	147
<b>第 12 章 変更履歴</b>	<b>149</b>
12.1 33.4.0 (未リリース) . . . . .	149
12.2 33.3.0 (2021/8/31) . . . . .	150
12.3 33.2.0 (2021/7/19) . . . . .	150
12.4 33.1.0 (2021/6/26) . . . . .	150
12.5 33.0.0 (2021/6/10) . . . . .	151
12.6 32.0.0 (2021/3/8) . . . . .	153
12.7 31.5.0 (2021/2/22) . . . . .	154
12.8 31.4.0 (2021/1/21) . . . . .	155
12.9 31.3.0 (2021/1/15) . . . . .	155
12.10 31.2.0 (2021/1/3) . . . . .	156
12.11 31.1.0 (2020/12/31) . . . . .	157
12.12 31.0.2 (2020/12/25) . . . . .	158
12.13 31.0.1 (2020/12/25) . . . . .	158
12.14 31.0.0 (2020/12/25) . . . . .	158
12.15 30.3.0 (2020/10/27) . . . . .	159
12.16 30.2.0 (2020/9/21) . . . . .	160
12.17 30.1.0 (2020/8/31) . . . . .	160
12.18 30.0.0 (2020/8/12) . . . . .	161
<b>第 13 章 既知の不具合</b>	<b>163</b>
13.1 「トラブルシューティング」に記載 . . . . .	163
13.2 対応予定なし . . . . .	163
13.3 現在進行系 . . . . .	164
13.4 既知の不具合だったもの . . . . .	164
<b>第 14 章 macOS で MMD をつかう方法</b>	<b>169</b>
14.1 BootCamp をつかう . . . . .	170
14.2 仮想マシンをつかう . . . . .	170
14.3 Unity + MMD4Mechanim をつかう . . . . .	171
14.4 Wine をつかう . . . . .	172
<b>第 15 章 Open Source Libraries</b>	<b>175</b>
15.1 Internal Libraries . . . . .	177
15.2 ffmpeg build configuration . . . . .	177
15.3 ICU build configuration . . . . .	178

第 16 章 nanoem の技術的説明	181
16.1 nanoem . . . . .	181
16.2 emapp . . . . .	186
16.3 よろずのおはなし . . . . .	194



**重要:** nanoem itself has English-ready UI however it's manual is not translated and there is no plan to translate it due to high maintenance cost. If you want to read English manual, translate this manual with [Google Translate](#) or use [translate webpages feature in Google Chrome](#) (At least the minimum quality to understand this manual is gualantieed).

---

nanoem は [MikuMikuDance](#) (以下 MMD) で使われるモデルやモーションの読み込み、及び編集と保存が可能な MMD 互換実装のアプリケーションです。

nanoem では MMD との互換性をもつことと同時に以下を目標として開発しています。

- 移植性
  - 主目標である macOS 以外に Windows で動作
  - 未公開であるものの Linux / Raspberry Pi / Windows10 on ARM64 上でも動作<sup>\*1</sup>
- 立ち上がりが軽量であること
  - 1 秒程度で立ち上がる<sup>\*2</sup>
- サイズが小さいこと
  - 10MB 程度<sup>\*3</sup>

---

<sup>\*1</sup> もし必要の場合は [不具合報告の方法](#) でお問い合わせください

<sup>\*2</sup> MMD と同じようにというのはありますがエディタ感覚で使えることを目標としています

<sup>\*3</sup> macOS における圧縮時のサイズ。31.0 以降はユニバーサルバイナリ化のためおよそ倍のサイズになっています



# 第 1 章

## インストール

### 1.1 必要な動作環境

#### 1.1.1 macOS 版

ご用心:

- CPU を最大限に使う（悪く言えば酷使用する）特性のため MacBook 系列上で使うとバッテリー駆動の場合はバッテリーが急激に減り、かつファンがうるさく鳴ります
- macOS 10.13.4 から外付け GPU に対応していますが nanoem では動作未確認です

- macOS 10.13 以上かつ 64bit 環境 (31.0 から Apple Silicon 対応)<sup>\*1</sup>
  - Metal が動作する環境を確認してください
  - 10.12 以下であっても 10.9 以上であれば動かしますが主にハードウェア的な問題で満足に動かせない可能性が高いためサポート対象外とさせていただきます
- 1280x800 以上の画面

#### 1.1.2 Windows 版

---

**重要:** Windows 版 nanoem は DirectX 11 上で動作します。そのため MMD や関連ツールで必要とする DirectX 9 ランタイムのインストールは不要です

---

---

<sup>\*1</sup> nanoem も例に漏れず MacBook において Intel 版よりも同サイズの Apple Silicon 版の方が明確に高い性能を発揮します。事実として高解像度モードでも重量級ポストエフェクトである Ray-MMD を 60FPS で動作させることができます

- Windows7 以上かつ 64bit 環境<sup>\*2</sup>
  - Visual Studio 2017 の Visual C++ 再頒布可能パッケージのインストールが必要
- 1280x800 以上の画面

### 1.1.3 Linux 版

---

注釈: 現状は [ソースからビルド](#) のみに限られます

---

- 64bit 環境
- OpenGL 3.3 以上または OpenGL ES 3.0 以上
- 1280x800 以上の画面

### 1.1.4 Retina 含む高解像度環境について

nanoem は Retina のような高解像度環境に対応していますが、高解像度モードは負荷が大きいため重いと感じた場合

- macOS 版は「[低解像度モード](#)」で開いてください。
- Windows 版はディスプレイ設定の拡大率 (DPI) を落として実行してください。

## 1.2 利用条件

この利用条件は予告なく変更されることがあります。

- nanoem を利用したことによって生じる一切の請求、損害、義務の責任を開発者及び開発貢献者は負わないものとする
  - nanoem は [Mozilla Public License 2.0](#) に基づく形で [ソースコードが公開](#) されています
- [プライバシーポリシー](#) に同意すること
- (過去に再配布禁止規定があったものの現在は削除)

---

<sup>\*2</sup> N/KN エディションを利用している場合は Windows のバージョンに応じた Media Feature Pack の追加インストールが必要です



## 1.3 ダウンロード方法

重要:

- 事前に上記の「利用条件」に同意したものとみなします
- 今後ダウンロード方法が変更される可能性があります
- 古いバージョンからアップグレードする場合で巻き戻しをしたい場合は nanoem 自体のバックアップをとってください

- macOS 版の配布ページ または Windows 版の配布ページ にアクセス
- *Download* ボタンを押してダウンロード
- zip ファイルを解凍
- Applications フォルダに nanoem をドラッグ・アンド・ドロップ    macOS 版のみ
  - 管理者権限が求めるダイアログが出ます
  - もし自身が管理者ではない場合はホームフォルダの「アプリケーション」フォルダにドラッグ・アンド・ドロップしてください
    - \* ホームフォルダに「アプリケーション」フォルダが存在しない場合は当該フォルダを作成してください
- nanoem をクリックして起動
  - 開かない場合は *nanoem* が開かない、起動しない にて

ご用心: macOS 版において初回実行時に開発元未確認による警告ダイアログが出ますがこれは仕様です。詳細は「Mac で App を安全に開く」の「ノータリゼーションを受けていない App や未確認の開発元の App を開く方法」を確認してください。

## 1.4 インストール後について

### 1.4.1 ヘルプがほしいとき

Twitter 上で [#nanoem 互助会](#) タグをつけてツイートすると誰かが拾ってくれるかもしれません。ただし互助会タグは開発者が介入しない方針のため開発者に直接聞く場合は後述の不具合報告の方法に記載しているマシュマロを利用してください。

Facebook とか LINE とかの場合は知ってそうな友達に聞くしかありません。

### 1.4.2 不具合報告の方法

まずは「よくある質問と回答」または「トラブルシューティング」を確認してから「マシュマロ」に送ってください。送る時間帯は気にせずいつでも送ってよいです。内容次第では「nanoem 不具合問い合わせ」にお願いすることがあります。

---

注釈:

- マシュマロはサービスの性質上書き方が悪いと本人のもとに届かないことがあります。また、届いたかどうかについての回答はしません
  - 有料のチョコマシュマロは対応していません
- 

### 1.4.3 不具合報告時の注意

- 以下の内容であれば開発者的に調査しやすいです。これらは現象を再現するために必要なファイルをお願いすることがあります。
    - プロジェクトを保存したがそのファイルが読み込めない
    - 画像または動画書き出ししようとしたが何回やっても落ちる
    - アクセサリ/モデル/モーションを読み込もうとしたがエラーが出る
- 

重要: 以下の場合には不具合対応しません

- 物理演算有効時にモデルが貫通する系の問題
  - macOS 10.9 (Mavericks) から 10.12 (Sierra) までの環境で動かしたときに発生する問題
  - macOS における OpenGL レンダラ 使用時のエフェクト利用全般
    - エフェクトを利用する場合は Metal に切り替えてから利用してください
- 

警告:

- 内容が「開発者から見て」あまりに粗雑だと対応しません
  - 例えば なんかクラッシュするけど対応して とか

- bowlroll のページにあるコメントを書いても返信しません

#### 1.4.4 スクリーンショットの撮り方

文章での説明が難しいものはスクリーンショットをとってください。撮り方は以下があります。

- 画像の場合は [Mac でスクリーンショットを撮る方法](#)
- 動画の場合は [画面を収録する](#)
  - [RetroClip](#) を使うと Nintendo Switch のように後から画面収録することが出来ます



## 第 2 章

# 逆引き利用ガイド

使う目的に合わせた利用方法の説明を行います。

### 2.1 とりあえず何をすればいいの？

---

ヒント：基本的な操作方法は [3Dミクを躍らせるツールを自作してみた（説明前編）](#) と [3Dミクを躍らせるツールを自作してみた（説明後編）](#) をみたほうがよいでしょう。2008 年のもののため UI が現在と当時のものとで異なりますが基本的な操作方法は変わりません。

---

最短な方法はモデルとそのモーションを読み込ませて再生することです。必要に応じてアクセサリを読み込みます。

#### 2.1.1 モデルを探す

[ニコニコ動画](#)、[ニコニコ静画](#)、[ニコニ立体](#) では「[MMD モデル配布あり](#)」タグを検索することで多数の、様々なモデルが見つかります。Twitter でも「[MMD モデル配布あり](#)」のハッシュタグで検索すると出てくることがあります。

多すぎて選択できない、という場合はニコニ立体で配布されている [アリシア・ソリッド](#) のモデルを使いましょう。

ご用心：配布されているモデルには取扱説明書と利用規約を兼ねる README またはそれに相当するファイルがあるので利用の際は必ず目を通すようにしてください。

---

ちなみに：nanoem では上記のテキストファイルを閲覧出来る機能を持っています。

---

### 2.1.2 モデルを踊らせる用のモーションを探す

ニコニコ動画、ニコニコ静画、ニコニ立体 では「MMD モーション配布あり」タグを検索することでモデルと同じく様々なモーションが配布されています。Twitter でも「MMD モーション配布あり」のハッシュタグ で検索すると出てくることがあります。

- モーションの場合は通常楽曲に合わせた形で配布されているため、ベースとなる楽曲がきまっていれば絞り込みをしやすいです
- 踊らせるモーションとは別にリップ（口パク）専用のモーションが配布されている場合があります

ご用心: 配布されてるモーションにはモデルと同じく取扱説明書と利用規約を兼ねる README またはそれに相当するファイルがあるので利用の際は必ず目を通すようにしてください。

---

ちなみに: nanoem では上記のテキストファイルを閲覧出来る機能を持っています。

---

### 2.1.3 アクセサリを探す

ニコニコ動画、ニコニコ静画、ニコニ立体 では「MMD アクセサリ配布あり」タグを検索することでモデルと同じく様々なアクセサリが配布されています。Twitter でも「MMD アクセサリ配布あり」のハッシュタグ で検索すると出てくることがあります。

- 「MMD ステージ配布あり」タグを検索することにより背景及び舞台となるアクセサリまたはモデルが見つかります。
- 「MMD スカイドーム配布あり」タグを検索することにより空背景となるアクセサリまたはモデルが見つかります。

---

注釈: スカイドームはステージを覆うくらい十分に大きい球状モデルに空のテクスチャが貼り付けられたものとして作られています。カメラをズームアウトするとそのことが確認できます。

---

### 2.1.4 カメラモーションを探す

「#MMD カメラモーション配布あり」で検索すると見つかります。カメラモーションはモデルを踊らせる方のモーションと一緒に配布されていることもありますが、通常は単体で配布されます。

カメラモーションはモデル（のモーション）の代わりにカメラを選択する形で読み込ませる方法は同じです。

## 2.2 モデルまたはアクセサリを読み込ませたい

- モデルは拡張子が *PMD* または *PMX* 形式のものを読み込むことができます。
- アクセサリは拡張子が *X* 形式のものを読み込むことができます。
  - アクセサリの場合は使用確認のダイアログが表示されないことを除いて方法は同じです。

### 2.2.1 方法その 1

詳細は「[モデル](#)」及び「[アクセサリ](#)」を参照

1. メニューの「ファイル」>「開く」>「モデル」を選択（アクセサリの場合は「ファイル」>「開く」>「アクセサリ」）
2. ダイアログが表示されるので読み込ませたいモデルまたはアクセサリのファイルを選択
3. 使用確認のダイアログが表示されるので内容を確認したら「OK」をクリック

### 2.2.2 方法その 2

1. 読み込ませたいモデルのファイルをドラッグ
2. nanoem のウィンドウに対してドロップ
3. 使用確認のダイアログが表示されるので内容を確認したら「OK」をクリック

### 2.2.3 方法その 3

1. 「モデル」または「アクセサリ」パネルの「読込」ボタンをクリック
2. ダイアログが表示されるので読み込ませたいモデルまたはアクセサリのファイルを選択
3. 使用確認のダイアログが表示されるので内容を確認したら「OK」をクリック

## 2.3 モーションまたはポーズを読み込ませたい

モーションは MMD で使われる VMD 形式または nanoem 独自の NMD 形式を読み込むことができます。

ポーズは MMD で使われる VPD 形式を読み込むことができます。また、モーフの情報が含まれる拡張形式のポーズファイルにも対応しています。

ご用心:

- モーションを読み込ませるときはフレーム位置を 0 にしてください (現在のフレーム位置が開始地点として認識されるため)
- NMD 形式は nanoem 独自の拡張モーション形式のため、他のアプリケーションでは読み込めません

### 2.3.1 モデル (ポーズを含む)

モデルのモーションまたはポーズはあらかじめ読み込ませたいモデルを選択してから読み込ませる必要があります。モデル読み込み直後であればその読み込まれたモデルが選択された状態となります。

方法その 1

詳細は「[モデルモーション](#)」を参照

1. モーションを適用したいモデルをメニューの「プロジェクト」>「選択」>「モデル」から選択
2. メニューの「ファイル」>「開く」>「モーション」>「モデルモーション」を選択 (ポーズの場合はメニューの「ファイル」>「開く」>「ポーズ」を選択)
3. ダイアログが表示されるので読み込ませたいモーションまたはポーズのファイルを選択

方法その 2

1. あらかじめモーションを読み込ませたいモデルを選択する
2. 読み込ませたいモーションまたはポーズのファイルをドラッグ
3. nanoem のウィンドウに対してドロップ

### 2.3.2 カメラ

詳細は「[カメラモーション](#)」を参照

1. メニューの「ファイル」>「開く」>「モーション」>「カメラモーション」を選択
2. ダイアログが表示されるので読み込ませたいカメラモーションのファイルを選択

### 2.3.3 光源

詳細は「[光源モーション](#)」を参照



1. メニューの「ファイル」>「開く」>「モーション」>「光源モーション」を選択
2. ダイアログが表示されるので読み込ませたい光源モーションのファイルを選択

## 2.4 モデルを動かしたい

---

ヒント: ここでは基本的な操作方法のみ解説しています。モーション作成につながる応用的な話は「[MMD モーション講座](#)」や「[MMD トレース講座](#)」タグがつけられている動画を確認するとよいでしょう。

---

動かす前にあらかじめモデルを選択する必要があります。まずモデルの選択方法について

### 2.4.1 モデルの選択

方法その1

1. メニューの「プロジェクト」>「選択」>「モデル」を選択
2. モデル名をクリック

方法その2

1. 「モデル」パネルの「デフォルト」をクリック
2. モデル名をクリック

動かしたいモデルを選択したら動かしたい対象をきめます。モデルは「ボーン」と「モーフ」の2つを動かすことによってモデルに動きを加える事が出来ます。静止状態はポーズとして、異なるポーズを複数して動きを形成したものがモーションとなります。

ボーンを動かすことによってモデルに柔軟な動きをさせることが可能ですが特性を理解しないと意図しない動きを起こしてしまうことがあるので最初はスライダーのみで動きを確認できるモーフを動かしたほうが良いかもしれません。

### 2.4.2 ボーンを動かす

以下「ボーン選択」と「ボーンハンドルの操作」と「ボーンのコピー及びペースト」にわけます

## ボーンを選択

ご用心: タイムライン上にあるキーフレームの  $\delta\check{\text{T}}\check{\text{u}}$  を選択してもボーンを選択したことになります。

いずれかの方法でボーンを選択することが出来ます。ボーンが選択されるとアイコンが赤く表示されます。

- モデル上の丸または四角アイコンをクリック
- タイムラインからボーン名が表示されている部分をクリック
- 「モデル」 > 「ボーンを選択」からボーン名をクリック

まとめて選択したい場合は [ボーンパネル](#) の「[箱選択](#)」を利用します。

- 「箱選択」をクリック
- ドラッグして範囲指定

## ボーンを動かす

ボーンはモデルを動かす要素の一つです。ボーンを動かすには以下のふたつの方法を利用します。

- 「ボーン」パネルの「回転」または「移動」ボタンをクリックすることで現れるハンドル
  - 「[回転モード \(ボーン\)](#)」ボタンで現れるハンドルは赤、緑、青で色分けされる (黄色は中心点だけで操作不可)

対象	何が起こるか
縦軸の赤	X 軸方向を動かす
円の外周の緑	Y 軸方向を動かす
横軸の青	Z 軸方向を動かす

- 「[移動モード \(ボーン\)](#)」ボタンで現れるハンドルは赤、緑、黄色で色分けされる

対象	何が起こるか
横軸の赤	「カメラから見て」X 軸方向に動かす
縦軸の緑	「カメラから見て」Y 軸方向に動かす
丸い黄色	「カメラから見て」任意の方向に動かす

- [ビューポート](#) 画面右下の X/Y/Z が並ぶアイコン
  - 上下左右の矢印があるアイコンがボーンを移動させるハンドル

- 回転してるアイコンがボーンを回転させるハンドル
- 両方共ドラッグして上下方向で動かす
- \* 左右には動かすことができないため注意

ヒント: ボーンを動かすとアイコンの色が青から緑に変化します。この状態で「未選択」ボタンをクリックすると自動的にそのボーンが選択されます。

操作が終わったら「ボーン」パネルの「登録」をクリックすることでキーフレームに登録します。

### ボーンのコピー及びペースト

ご用心: タイムラインにある「コピー」及び「ペースト」はキーフレーム用に独立した機能のため、押しても機能しません。また、メニューの「編集」にある「コピー」あるいは「ペースト」を利用するときは選択モード(ボーン)である必要があります。

注釈: ペースト時は異なるモデルにも適用することが可能ですが、その場合はボーン名が一致している必要があります。

1. 「ボーンを選択」の方法でボーンを選択
2. メニューの「編集」>「コピー」あるいは ボーンパネルにある「コピー」でコピー
3. コピーする先のフレーム位置に移動
4. メニューの「編集」>「ペースト」あるいは ボーンパネルにある「ペースト」でペースト

### 2.4.3 モーフを動かす

モーフはボーンに並びモデルを動かす要素のひとつで、以下のカテゴリ分けされています。

カテゴリ名	説明	例
唇	主に口の動きを制御する	あ、い、う
目	主に目の動きを制御する	まばたき、ウィンク
眉毛	主に眉毛の動きを制御する	真面目、困る
その他	上記のいずれにも所属しない	瞳小

ご用心: 登録ボタンがカテゴリごとにそれぞれありますが処理が独立しています。例えば「目」カテゴリのモーフを動かして「その他」のカテゴリの登録ボタンを押しても「目」カテゴリではなく「その他」で選択されているモーフが登録されてしまいます。

---

ヒント: 左右のボタンをクリックすると 0.0 から 1.0 の間、0.01 単位で調整出来ます。また、数値の部分をダブルクリックすると直接数値入力することが出来ます。

---

1. モーフを動かしたいモデルを読み込み、選択する
2. 「[モーフパネル](#)」からモーフ名の下の「ウェイト」または「ウェイト」と数値の空白部分を左右にドラッグ
3. モーフの値を調整し終わったら「登録」をクリックしてキーフレームを登録
4. 1. の繰り返し

## 2.5 モーションのキーフレームをコピー＆ペーストしたい

---

ヒント: 縦方向ですべてのキーフレームを選択したい場合は「[列選択](#)」を押します。

横方向でまとめてキーフレームを選択したい場合は [ボーンパネル](#) の「[選択](#)」を押します。こちらは From と To で選択範囲を設定することができ、左隣のプルダウンですべて選択するか特定のボーンあるいはモーフだけ選択することが出来ます。

---

ご用心: [ボーンパネル](#) にある「コピー」及び「ペースト」はボーン用に独立した機能のため、押しても機能しません。また、メニューの「編集」にある「コピー」あるいは「ペースト」を利用するときは [選択モード \(ボーン\)](#) 以外 である必要があります。

1. マーカーを選択または [タイムライン](#) 上をドラッグしてキーフレームを選択
2. メニューの「編集」>「コピー」あるいは [タイムライン](#) にある「[コピー](#)」でコピー
3. コピーする先のフレーム位置に移動
4. メニューの「編集」>「ペースト」あるいは [タイムライン](#) にある「[ペースト](#)」でペースト

## 2.6 音源を読み込ませたい

---

注釈: うまく読み込めない場合は「音源が読み込めないあるいは音がでない」を確認してください。

---

1. メニューの「ファイル」>「開く」>「音源」をクリック
2. 読み込ませたい音源のファイルをクリック
  - macOS 版は以下のファイル拡張子の読み込みに対応しています
    - wav
    - mp3
    - aac

音源が読み込まれるとフレームの最大値が音源の再生時間に合わせて自動的に調整されます。

### 2.6.1 音源を一回消去したい

音源を消去したい場合メニューの「プロジェクト」>「音源を消去」で可能です。その際再生時間が自動的に調整されます。

## 2.7 背景動画を読み込ませたい

以下の手順で可能です。別の背景動画を再読込すると前のものと入れ替わる形で読み込まれます。

1. メニューの「ファイル」>「開く」>「背景動画」をクリック
2. 読み込ませたい背景動画のファイルをクリック

---

ヒント: シフトキーを押しながら画面右上のカメラアイコンをドラッグすると背景動画の拡張及び位置を動かすことが出来ます。これは主にモーショントレース向けの機能となります。

---

詳しくは「[背景動画](#)」を確認してください。

### 2.7.1 背景動画を一回消去したい

背景動画を消去したい場合メニューの「プロジェクト」>「背景動画を消去」で可能です。

## 2.8 再生したい

再生パネルの「再生」ボタンをクリックすることで再生できます。

メニューの「プロジェクト」>「再生」で再生することも出来ます。

### 2.8.1 ループ再生及び範囲再生

ループさせたい場合は同じく「再生」パネルの「ループ」にチェックを入れることでループ再生が可能となります。また、フレームの範囲を指定したい場合は「開始」と「終了」にそれぞれのフレーム位置を指定した上で左側のチェックボックスにチェックを入れてください。

## 2.9 エフェクトを読み込ませたい

エフェクトはモデルまたはアクセサリと同時に読み込み、適用する仕組みとなっています。方法については「[モデルまたはアクセサリを読み込ませたい](#)」を参照してください。拡張子は FX または FXN 形式のものを読み込むことが出来ます。

---

注釈: 1.14.0.0 からエフェクトを読み込むときエフェクトプラグインが無効の場合確認ダイアログが表示されるようになりました。

- OK ボタンをクリックするとエフェクトプラグインを有効に切り替えた上でエフェクトが自動的に読み込まれます。
- Cancel ボタンをクリックすると読み込み自体がキャンセルされます。

1.12.0.0 から拡張子が FX のものも読み込めるようになりました。

- 原理的には MME のエフェクトをそのまま読み込むことが出来るようになります。
  - 読み込まれなかったり正しく表示されないことがあります。
  - エフェクトプラグインと呼ばれる仕組みを使って読み込む形で、デフォルトは無効です。
- 

## 2.10 アクセサリまたはモデルを特定のモデルに持たせたい (外部親)

アクセサリまたはモデルを特定のモデルに持たせる機能として外部親があります。これを利用することにより例えば武器を特定のモデルに持たせることが可能になります。

### 2.10.1 アクセサリを持たせる

1. カメラ画面に切り替える（モデルを選択しない状態にする）
2. [アクセサリパネル](#) から持たせたい対象のアクセサリを選択する
3. [外部親（アクセサリ）](#) のボタンを押す
4. 「接続先親モデル」と「接続先ボーン」を設定する
5. 「外部親登録」を押す

### 2.10.2 モデルを持たせる

アクセサリと異なり対象ボーンを追加で設定する必要がありますが、ひとつのモデルで複数のボーンを外部親として接続することが可能です。これは複数のアクセサリを同一の接続先親モデルに設定する場合モデルの外部親の方が有利に働きます。

1. 持たせたい対象のモデルを選択する
2. [モデルパネル](#) にある [外（外部親設定）](#) のボタンを押す
3. 「対象ボーン」と「接続先親モデル」と「接続先ボーン」を設定する
4. 「外部親登録」を押す

## 2.11 物理演算の設定を変えたい

「プロジェクト」>「物理演算」>「演算設定」で設定可能です。詳しくは「[演算設定](#)」を確認してください。

## 2.12 IK/物理演算を無効化したい

IK 及び物理演算は操作の簡易化及び動きのリアリティを上げることに貢献しますが、複雑さ故に意図しない動きが発生することがあります。その場合はあえて IK または物理演算を無効化して手で動きをつけたいときがあります。

ここでは IK または物理演算を無効化する方法を記載しています。

### 2.12.1 IK を無効化する

以下の手順で IK を無効にすることができます。

1. 無効化したいフレーム位置に移動

2. モデルパネルの「外」ボタンの左隣にあるリストから無効化したいIK 対象を選択
3. その下の ON/OFF から OFF をクリック
4. その下の「登録」をクリック

一回無効にすると次のモデルのキーフレームが有効になるまで無効のままになります。

### 2.12.2 ボーン単位で物理演算を無効化する

ご用心: ボーンパネルの「物理」のチェックボックスはキーフレーム登録時に利用するもののため、現在位置のキーフレームにおける物理有効無効の状態を反映しません。タイムライン上のマーカーの方を確認してください。

以下の手順でボーン単位での物理演算を無効化出来ます。

1. 無効化したいフレーム位置に移動
2. 無効化したいボーンを選択
3. ボーンパネルにある「物理」のチェックを外す
4. その下の「登録」をクリック

タイムライン上のマーカーが  $\text{a}\dot{\text{i}}\ddot{\text{N}}$  から  $\text{o}\S\ddot{\text{T}}\ddot{\text{u}}$  に変化していれば無効化できています。特性上ややこしい仕様があるため詳細は「物理」を確認してください。

## 2.13 準標準ボーンを入れたい

---

注釈: 配布されるモデルはよほど古いモデル出ない限りは準標準ボーンが予め入っています。一方で自作モデルや VRoid Studio から変換したものなどについては入っていないため準標準ボーンを入れる場合は実行が必要になります。

---

まず **プラグイン** を確認した上で「**準標準ボーンプラグイン**」の導入を行ってください。PMDEditor/PMXEditor のそれとは互換性がなく入れても機能しないため注意が必要です。

適用対象のモデルを選択した上で「編集」>「モデルプラグイン」>「準標準ボーンプラグイン」>「設定と実行」を選択し、必要に応じて設定を行った上で「OK」をクリックしてください。



## 2.14 モーショントレースしたい

警告: 初心者がいきなりやるには無謀で挫折する可能性が極めて高いので少なくとも以下ができてからにしましょう。

- モデルを読み込める
- モーションを読み込める
- カメラを動かせる
- ボーンを動かせる
- モーフを動かせる
- 背景動画の読み込み方法と背景動画の移動方法を理解した
- ポーズが作れる
- キーフレームのコピー＆ペーストができる
- 「[標準ボーンプラグイン](#)」が扱える

またモーショントレース元の「踊ってみた」系の振り付けに著作権が発生し得るため<sup>\*1\*2</sup> 外部に公開する際はモーショントレース元の許可を事前にとってください。

基本的には背景動画を読み込み、モデルを背景動画に合わせてポーズを作り上げて次のコマに移動してまたポーズを作り上げる... を繰り返すのみです。明確なゴールはあるものの時間と忍耐力に多大なコストをかける作業です（慣れの部分があるので慣れで作業時間を短縮できることはあります）。

nanoem に直接背景動画を読み込ませず、動画プレイヤーに動画を読み込ませて参照動画として使う方法もありますがどちらが最適かは個々人にまかせます。

方法論については「[#MMD モーション講座](#)」または「[#MMD トレース講座](#)」タグで検索してください。

## 2.15 その他よく聞かれる質問について

よくある質問と回答 を参照（問題が起きた場合は [トラブルシューティング](#)）

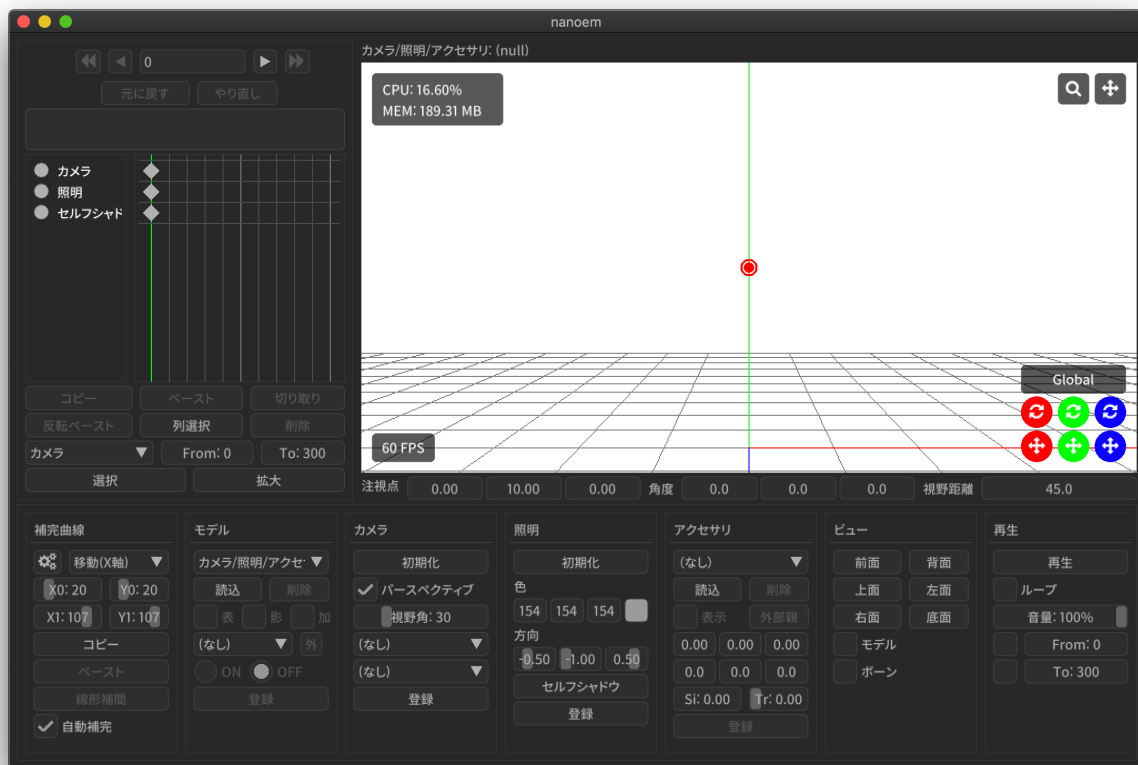
<sup>\*1</sup> <https://copyright-topics.jp/topics/dance/>

<sup>\*2</sup> <https://monolith-law.jp/youtuber-vtuber/tried-to-dance-copyright-infringement>



## 第 3 章

# アプリケーションの画面概要



大まかには以下の 3 つに大別されます。

- タイムライン
  - モデルやアクセサリなどを動かす要素であるモーションのキーフレームを可視化したもの
  - Blender や Unity その他 3DCG アプリケーションでいうところのドープシートに近い
- ビューポート

- モデルやアクセサリを表示する領域
- MikuMikuDance ではプレビュー画面と呼ばれている
- パネル
  - モデルやアクセサリその他を操作する領域

---

ちなみに:

- 入力欄によっては左右にドラッグすることによって数値変化するものがあります
  - スライダーはシフトを押しながらクリックすると数値入力することができます
- 

## 3.1 タイムライン



### 3.1.1 (リサイズ)

---

注釈: この機能は 26.4 から導入されました

---

タイムラインと **ビューポート** の間にカーソルを当てると緑色のバーが表示されます。これをドラッグするとタイムラインのリサイズができ、**ビューポート** もタイムラインにあわせて拡大縮小されます。

初期値はタイムラインと **ビューポート** の割合が 30:70 になるように調整され、 25:75 から 75:25 の間で調整可能

です。30:70 に近い割合でリサイズが行われようとした場合自動的に 30:70 になるように自動的にリサイズ（スナップ）されます。

ご用心:

- 既知の問題として何回もリサイズが行われた場合 **ビューポート** が徐々に小さくなる問題があります
- その場合は「ビューポート設定ダイアログ」から一度解像度の値を変更してもとに戻してください
- ウィンドウを拡大縮小しても効果はないため上記手段で対処をお願いします
- スナップを切る機能はありません

### 3.1.2（フレーム移動）

現在のフレーム位置を移動します

重要:

- 「フレーム」は「キーフレーム」とは別です！
  - 具体的にいうとキーフレームはタイムライン上に  $\delta\text{\textcircled{S}}\text{\textcircled{T}}\text{\textcircled{U}}$  のマークが表示されてます
  - 選択されたキーフレームは  $\delta\text{\textcircled{S}}\text{\textcircled{T}}\text{\textcircled{U}}$  のマークの色がオレンジ色になります
- 右方向のボタンをクリックすると前のフレームに移動します
  - 例えば現在のフレーム位置が 123 の場合押すと 124 に移動する
- $\text{\textcircled{A}}\text{\textcircled{R}}\text{\textcircled{L}}$  二重右方向のボタンをクリックすると前方向のキーフレームが存在するフレームに移動します
  - 例えば現在のフレーム位置が 123 で次のキーフレームが 115 にある場合は 115 に移動します
  - 主にカメラモーションのキーフレーム移動で利用します
- 左方向のボタンをクリックすると後ろのフレームに移動します
  - 例えば現在のフレーム位置が 123 の場合押すと 122 に移動する
- $\text{\textcircled{A}}\text{\textcircled{R}}\text{\textcircled{L}}$  二重左方向のボタンをクリックすると後ろ方向のキーフレームが存在するフレームに移動します
  - 例えば現在のフレーム位置が 123 で次のキーフレームが 130 にある場合は 130 に移動します
- ボタンは押し続けることができます
  - 長押しすることで高速に移動することができます

- 数値入力が可能で、エンターキーを押すとその位置に移動します

### 3.1.3 (タイムライン内の範囲ドラッグ)

- タイムライン内はドラッグすることができます
  - ドラッグ中はドラッグ開始位置から終了位置まで赤い四角で表示されます
  - 赤い四角の中に含まれたキーフレームは選択状態になります

### 3.1.4 (キーフレームのドラッグ)

---

注釈: この機能は 25.0.0 から導入されました

---

- キーフレームを選択した状態でドラッグすると選択されたキーフレームを移動させることができます
  - フレーム位置が 0 番目にあるキーフレームはモーションの仕様により削除ではなく複製する形で移動します
  - 移動先にキーフレームが存在する場合は移動先に存在するキーフレームは削除される形で上書きされます

### 3.1.5 元に戻す

元に戻す と同じです

### 3.1.6 やり直し

やり直し と同じです

### 3.1.7 (トラック)

- モデルが選択されている場合は以下の順番で表示
  - 最初のボーン
  - モデル
  - 表情 (モーフ)

- \* カテゴリ分けされず全て表示される
- ボーン
  - \* モデルでカテゴリ分けがされている場合それに基づいて表示
- モデルが選択されていない場合は以下の順番で表示
  - カメラ
  - 光源

### 3.1.8 コピー

選択されたキーフレームをコピーします。

- [補完曲線パネル](#) や [ボーンパネル](#) にあるコピーとは独立しているためそれらと共有することは出来ません

### 3.1.9 ペースト

「コピー」されたキーフレームを現在のフレーム位置に対してペーストします。

- [補完曲線パネル](#) や [ボーンパネル](#) にあるペーストとは独立しているためそれらと共有することは出来ません
- コピーされない限りボタンをおすことは出来ません

### 3.1.10 反転ペースト

ご用心: 同一モデルの左右の手足の動きをあわせるために使うものなので、たとえばふたつのモデルを使って鏡合わせ的な用途で利用することはできません

「コピー」されたキーフレームを現在のフレーム位置に対して反転ペーストします。

- 「反転ペースト」は以下の特性を除いて「ペースト」と同じです
  - ボーン名の先頭に「右」または「左」が含まれている場合、それらの値を「左」または「右」になるように値を調整する
- [補完曲線パネル](#) や [ボーンパネル](#) にあるペーストとは独立しているためそれらと共有することは出来ません
- コピーされない限りボタンをおすことは出来ません

### 3.1.11 列選択

選択されているキーフレームを全てのトラック（列であるため）に登録されているキーフレームに対して選択します。

列選択はひとつ以上キーフレームを選択する必要があります。キーフレームがひとつも選択されていない場合は何も起こりません

### 3.1.12 削除

---

注釈: キーフレームのうちフレーム位置が 0 のものは削除することが出来ません。削除しようとしてもフレーム位置が 0 のものは必ず残ります

---

選択されている全てのキーフレームを削除します。

### 3.1.13 開始

キーフレームを選択するフレーム位置の開始位置を設定します。

「開始」は「選択」と「拡大」でのみ使われます。

### 3.1.14 終了

キーフレームを選択するフレーム位置の終了位置を設定します。

「終了」は「選択」と「拡大」でのみ使われます。

### 3.1.15 （選択されたトラック）

トラックを選択します

「選択されたトラック」は「選択」と「拡大」でのみ使われます。

### 3.1.16 選択

上記の「開始」と「終了」の範囲内にあり、かつ「選択されたトラック」で指定されたキーフレームを全て選択します。



### 3.1.17 拡大

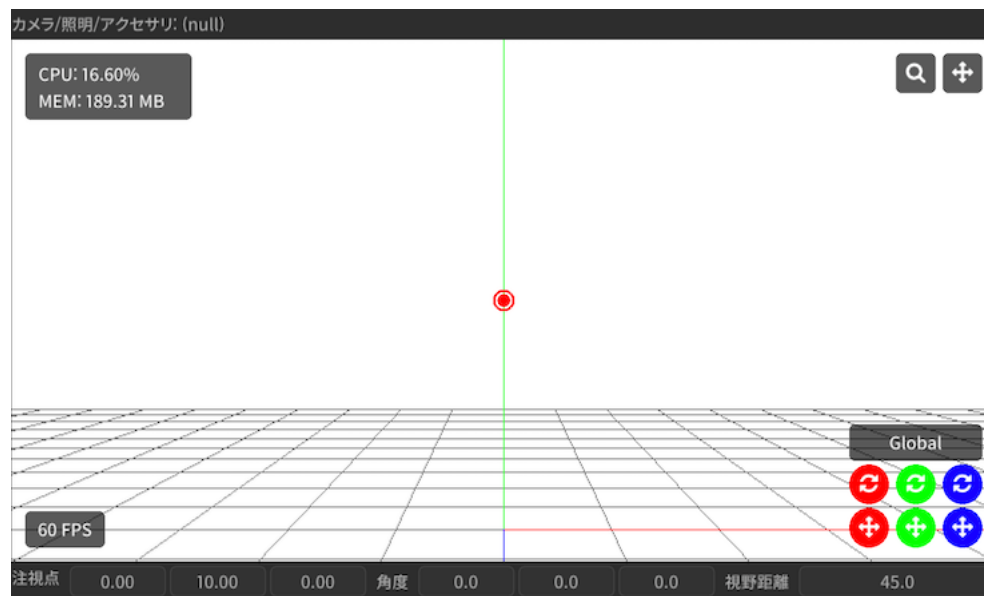
ご用心: カメラは隣接するキーフレームをシーン切り替えとして利用する仕様のため対応していません (MikuMikuDance でも同様の理由からできません)。かわりにカメラのキーフレームを手動でずらしてください

上記の「開始」と「終了」の範囲内にあるすべてのキーフレームのフレーム時間を 0.1-10 倍の単位で拡大または縮小します。

- 拡大した場合は処理された箇所がスローモーションのように再生されます
- 縮小した場合は処理された箇所が高速再生されます
- 選択範囲より後ろにあるすべてのキーフレームは拡大または縮小するフレーム時間分を自動的に移動します

カメラ以外のモーションで適用可能です。モデルの場合は選択されたモデルに対して実行するため、全てのモデルに適用する場合はお手数ですがモデルを選択して拡大縮小を全てのモデルに対して繰り返し実行してください。

## 3.2 ビューポート



- 画面左上には選択されているモデル及びボーンの名前が表示されます
  - モデルが選択されていない場合は カメラ/照明/アクセサリ が表示されます
  - モデルのボーンが選択されていない場合は (null) が表示されます
- CPU は現在の nanoem の CPU 消費率を示します

- MEM は現在の nanoem のメモリ消費量を示します
- FPS は現在の秒間フレーム描画数を示します
  - 60FPS を維持できれば安定して描画できています

---

ヒント:

- CPU と MEM の表示有無は [FPS カウンターの表示を有効にする](#) で切り替えられます
  - FPS の表示有無は [パフォーマンスモニターの表示を有効にする](#) で切り替えられます
- 

### 3.2.1 マウス操作

---

**重要:** MacBook Air または MacBook Pro のタッチパッド利用時のみ操作上の観点から右ボタンが 左ボタン に切り替わります

---

[ビューポート](#) においてマウス操作を行うことでカメラ操作またはモデル操作が可能です。ただし再生中は一切の操作を受け付けなくなりますので操作する場合は一回再生を停止させてください。

Undo/Redo ボタンは利用中のマウス（例えばロジケールの MX シリーズなど）に存在しかつ認識できる場合にのみ使えます。

#### カメラモードの場合

右ボタンでドラッグ	カメラを回転
中ボタンでドラッグ	カメラを移動
左ボタン	<a href="#">ビューポート</a> のハンドルに反応するのみ
Undo ボタンをクリック	巻き戻し
Redo ボタンをクリック	やり直し

モデルが選択されている場合

選択モード中に右ボタンでクリック	ボーンを選択
箱選択時に右ボタンでドラッグ	ボーンを範囲選択
回転モード中に右ボタンでドラッグ	選択されたボーンを回転
移動モード中に右ボタンでドラッグ	選択されたボーンを移動
中ボタンでドラッグ	カメラを移動 カメラモードとは独立
左ボタン	ビューポート のハンドルに反応するのみ
Undo ボタンをクリック	巻き戻し
Redo ボタンをクリック	やり直し

### 3.2.2 カメラのズーム

ビューポート 画面右上の虫眼鏡アイコンをドラッグするとカメラのズームを行うことができます。

- 上方向にドラッグするとズームイン
- 下方向にドラッグするとズームアウト

SHIFT キーを押しながら操作するとカメラのかわりに背景動画のズームが可能になります。

### 3.2.3 カメラの注視点移動

ビューポート 画面右上の4方向矢印アイコンをドラッグするとカメラの注視点を移動させることができます。

SHIFT キーを押しながら操作するとカメラのかわりに背景動画の位置移動が可能になります。

### 3.2.4 ハンドル

モデルやカメラを動かすためのもので、ビューポート 画面右下にある6つの丸いボタンとその上にあるボタンで構成されています。

- ドラッグすることによりモデルまたはカメラを移動または回転させることが出来る
  - 赤が X 軸、緑が Y 軸、青が Z 軸をそれぞれ表す
  - 4方向矢印のアイコンが「移動」、2つの矢印が回るように表示されるアイコンが「回転」
    - \* モデル操作時にアイコンにカーソルをあてると選択中のボーンに XYZ 軸が表示される
- Global または Local と表示されている部分をクリックすると変形基準を変えることが出来る
  - Global は XYZ 軸に沿った回転を変形を行う

- Local は基本的に Global と同じだが「ローカル軸」が設定されている場合その軸に沿った変形を行う

### 3.2.5 モード切替

ビューポート 画面左下にあるボタンで、カメラとモデルのそれぞれのモードを切り替えることができます。

- モデルがすでにあり、かつモデルが選択されていない場合は「モデル」ボタン
  - 押すと最後に選択されたモデルが選択されてモデルモードに切り替わる
- モデルが選択されている場合は「カメラ」ボタン
  - 押すとカメラモードに切り替わる

### 3.2.6 数値入力

ビューポート 画面下に占めているもの

- カメラモードの場合は以下の項目を数値入力することが可能
  - 注視点
  - 角度
  - 視野距離
- モデルモードの場合は以下の項目を数値入力することが可能
  - 移動
  - 回転
    - \* 移動あるいは回転不可に設定されている場合は読取専用になりグレースアウトされる
- 入力中は入力された数値に基づいて暫定的に移動または回転が反映される
- いずれも数値入力後、入力前の値が「元に戻す」の操作に記録される
- X/Y/Z ボタンをクリックするとリセットすることが出来る

## 3.3 補完曲線パネル

---

注釈: 補間曲線パネルはキーフレームがボーンまたはカメラの場合のみ機能します

---

補完曲線はキーフレームとキーフレームの間の動きの変化を設定するものです。補完曲線のメリットとして

- 1 フレームずつパラメータを設定してキーフレーム登録するよりも遥かに効率的に動きをつけることが出来る
  - あとから動きを調整したい場面において補完曲線のパラメータをいじるだけで済む
    - \* 1 フレームずつのキーフレーム登録だと影響する全てのキーフレームを修正する必要があるため非常に手間がかかる
- キーフレームを減らすことが出来ることからモーションファイルを小さくすることが出来る

が挙げられます。ただし大きな動きの変化があるものは補完曲線では制御が難しくなるため明示的にパラメータを設定してキーフレーム登録の方が有利かもしれません。

### 3.3.1 補間曲線グラフパネル

ご用心:

- 表示するためのギアアイコンはキーフレームが一つ以上選択されていないと有効になりません
- ボーンの場合、ボーンを選択しただけではボタンが無効のままなので注意が必要です

ギアアイコンをクリックすると選択された全てのボーンまたはカメラのキーフレームに対する補完曲線を描画したパネルが表示されます

- 選択されたキーフレームにそれぞれ異なる補完曲線が設定されている場合重ね合わせる形で描画されます
  - 「線形補間」の場合は名前が示すとおり左下から右上の斜め一直線が引かれます
  - X 軸が時間、Y 軸がキーフレームとキーフレームの間のパラメータの比率
    - \* X 軸は左が前のキーフレーム、右が次のキーフレーム
    - \* Y 軸は下が前のキーフレーム、上が次のキーフレーム
- ひとつもキーフレームが選択されていない場合は描画されません
- 1.19 から緑丸マークが表示され、今のフレーム位置が補間曲線上どこにいるか把握出来るようになっていきます

補間曲線を変更する

---

注釈: 補間曲線はキーフレームがひとつだけ選択してる場合のみ変更することが出来ます

---

補間曲線は二箇所ある丸い点をドラッグすることにより変更することが出来ます。

- 変更対象がボーンの場合は「ボーン」パネルの「登録」ボタンを押して反映させる必要があります
- 変更対象がカメラの場合は「カメラ」パネルの「登録」ボタンを押して反映させる必要があります

### 3.3.2 （対象の選択）

操作する補完曲線の対象を選択します

- カメラ
  - X 軸の注視点
  - Y 軸の注視点
  - Z 軸の注視点
  - 角度
  - 視野距離
  - 視野角
- モデル（ボーン）
  - X 軸の移動
  - Y 軸の移動
  - Z 軸の移動
  - 回転

### 3.3.3 コピー

選択されたキーフレームの補完曲線をコピーします 未実装

---

注釈: タイムラインや [ボーンパネル](#) にあるコピーとは独立しているためそれらと共有することは出来ません

---

### 3.3.4 ペースト

「コピー」された補完曲線を現在選択されているキーフレームに対してペーストします

- 未実装
  - 同名のもの、かつ「選択」で指定された対象に対してペーストする挙動になる予定

---

注釈: タイムラインや **ボーンパネル** にあるコピーとは独立しているためそれらと共有することは出来ません

---

### 3.3.5 線形補間

現在選択されているキーフレームの補完曲線を全て線形補間に設定します

## 3.4 カメラパネル



#### 3.4.1 初期化

初期化 と同じです

#### 3.4.2 パースペクティブ

カメラのパースペクティブ状態の有効無効を切り替えます

#### 3.4.3 視野角

カメラの視野角を調整します

- 左右のボタンで 1 度ずつ調整できます

- 空白部分をドラッグすると移動した分だけ適用されます
- 1 度から 135 度の間で調整可能です

### 3.4.4 （接続親モデルの選択）

カメラの視点先となるモデルを選択します。

単独では機能せず、接続親モデルのボーンを選択が必要です。

### 3.4.5 （接続親モデルのボーンを選択）

カメラの視点先となるモデルのボーンを選択します。

設定するとカメラの視点先がモデルのボーンの動きに合わせて追従ようになります。このため、カメラを直接操作するよりも接続親モデルを操作してカメラのキーフレームを登録する方法が場合によっては簡単な場合があります。

### 3.4.6 登録

現在のフレーム位置に対してカメラのキーフレームを登録します

## 3.5 照明パネル



### 3.5.1 （色選択）

光源光の色を設定します



### 3.5.2 （方向）

光源光の方向を設定します

- クリックすると X/Y/Z 軸それぞれのパラメータが表示されます
  - 左右のボタンで 0.01 ずつ増減調整出来ます
  - 空白部分をドラッグすると移動した分のパラメータが増減します

ご用心: 「登録」でキーフレームに登録されない限りフレーム切り替えで消失します

### 3.5.3 セルフシャドウパネル

セルフシャドウパネルを開きます。以下を設定することが可能です。

モード

距離

セルフシャドウがカバーする距離を設定します。距離を変えることによる挙動はモードによって変わります

- モード 1（バランス型）の場合
  - 小さくするとカメラに近いオブジェクトのセルフシャドウが細くなる
    - \* かわりに遠い部分のセルフシャドウが切り捨てられる
  - 大きくするとカメラから遠いオブジェクトのセルフシャドウがかかる
    - \* かわりにカメラに近いオブジェクトのセルフシャドウが荒くなる
- モード 2（近距離重視）の場合
  - モード 1 よりも小さくしたときのセルフシャドウがより細くなる
  - モード 1 と違って大小によってセルフシャドウが途切れるな挙動はない

初期化

セルフシャドウの設定を初期化します。

登録

現在のフレーム位置に対してセルフシャドウのキーフレームを登録します。

### 3.5.4 初期化

初期化 と同じです

## 3.6 モデルパネル



### 3.6.1 （モデルの選択）

---

注釈: 30.3.0 からモデルの名前変更機能が追加されました

---

---

重要: プロジェクト内で使われるモデル名なのでここでモデル名を変更しても読み込まれたモデルを直接書き換えるような変更はありません

---

一覧からモデルを選択します。

コントロールキーを押しながらクリックするとプロジェクト内のモデルの名前変更ができます（エンターキーを押すとモデル選択に戻ります）。同じモデルを読み込ませた場合で名前を区別するときに利用してください。

### 3.6.2 読込

モデル と同じです

### 3.6.3 削除

現在選択中のモデルを削除 と同じです

### 3.6.4 可視

モデルの表示と非表示を切り替えます。

- モデルを非表示すると付随するエッジや影も非表示になります
- カメラから外れてて見えないモデルは非表示にしたほうが描画速度が向上します
- 「登録」でキーフレームに登録されない限りフレーム切り替えで消失します

### 3.6.5 影

モデルに付随する影の表示と非表示を切り替えます。

### 3.6.6 加算

### 3.6.7 (IK ボーンの選択)

IK ボーンを選択します。IK ボーンの選択状態は *On* と *Off* ボタンの切り替えのみ使われます

### 3.6.8 外 (外部親設定)

---

注釈: プロジェクト及び NMD にはモデルのキーフレームとして記録されますが、技術的仕様で VMD には記録されません

---

モデルの外部親を設定します。

この機能は接続先モデルとそのボーンを設定することにより元モデルの位置及び回転の計算を自動的に行なってくれます。これにより例えば接続先親モデルに装備品となるモデルをもたせるような動きで必要なモデルの位置及び回転の計算を肩代わりしてくれます。

同様の機能は「アクセサリ」パネルの「外部親 (アクセサリ)」がありますが、こちらはアクセサリ専用です。

#### 対象ボーン

接続元モデルの基準となるボーンを設定します。

#### 接続先親モデル

---

注釈: 自分自身 (= 接続元モデルと接続先親モデルが同じ) にも設定できます

---

追従する外部親のモデルを設定します。

#### 接続先親ボーン

追従する外部親のモデルのボーンを設定します。

#### 外部親登録

「対象ボーン」、「接続先親モデル」及び「接続先親ボーン」を元にモデルのキーフレームを登録します。

### 3.6.9 On

選択された IK ボーンを有効にします

ご用心: 「登録」でキーフレームに登録されない限りフレーム切り替えで消失します

### 3.6.10 Off

選択された IK ボーンを無効にします

ご用心: 「登録」でキーフレームに登録されない限りフレーム切り替えで消失します

### 3.6.11 登録

モデルの可視状態と IK ボーンの状態をキーフレームに登録します。

## 3.7 ボーンパネル

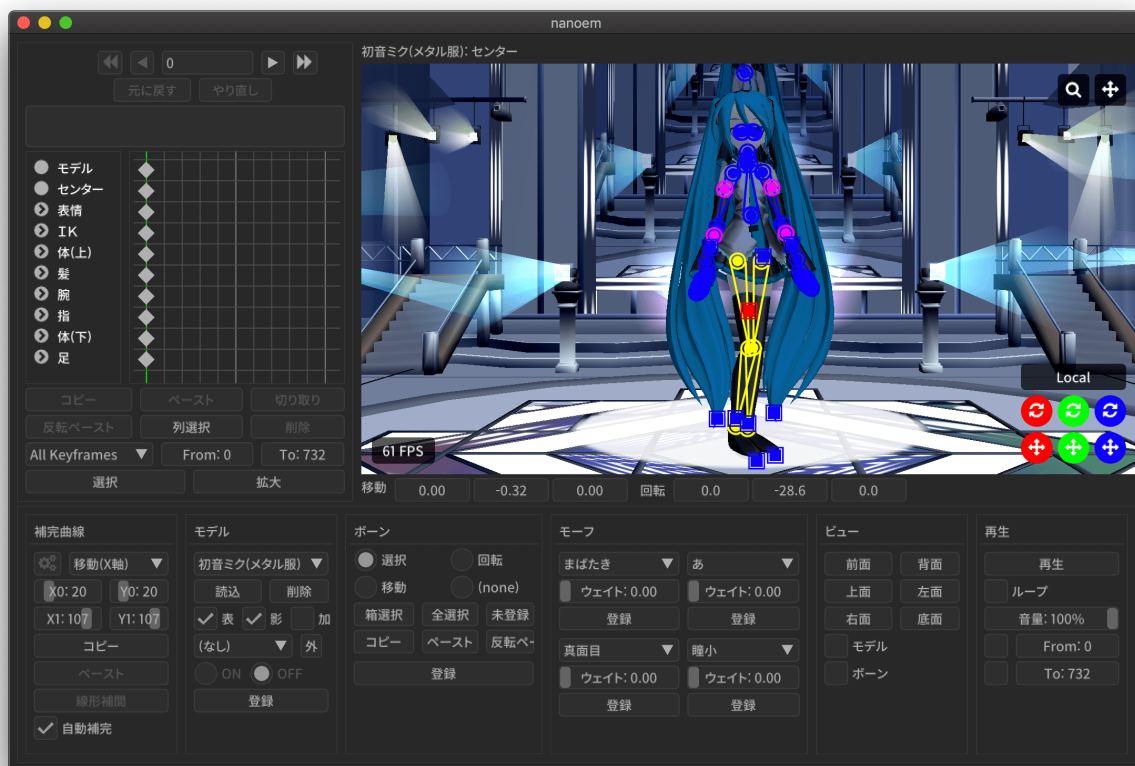
---

注釈: モデルが選択されている場合のみ表示されます

---



### 3.7.1 選択モード（ボーン）



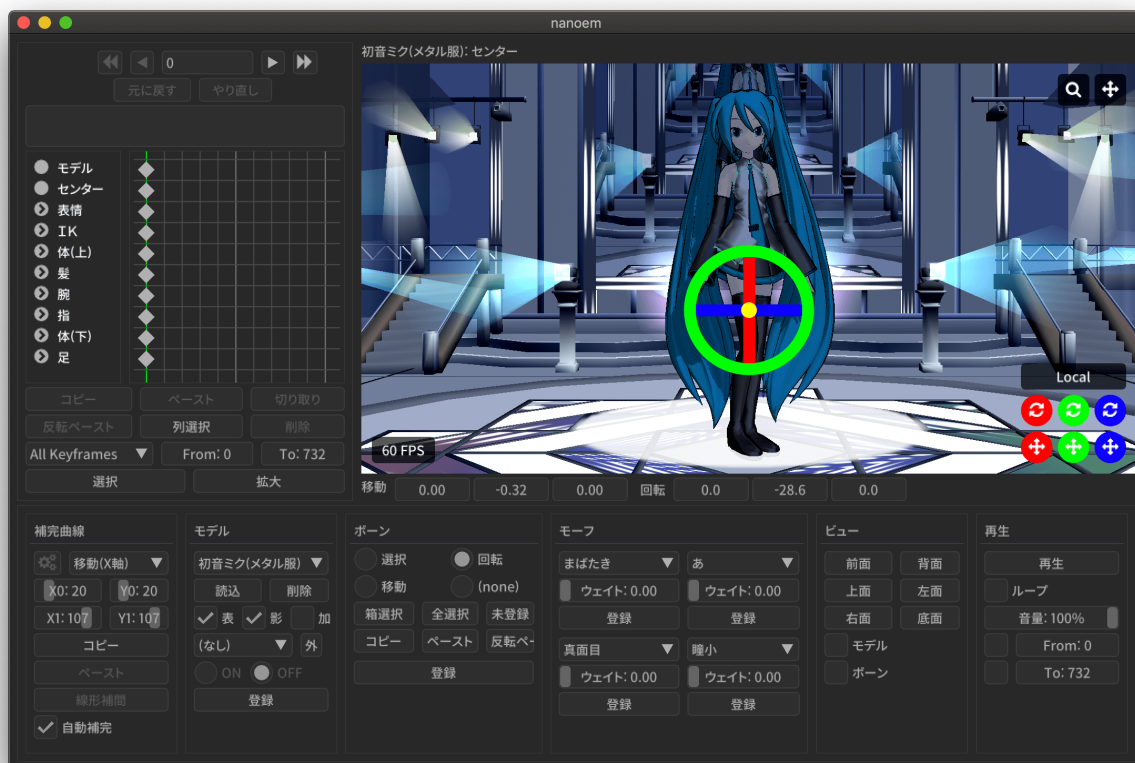
選択モードに切り替えます。

- ボーンを中心点とそれぞれを接続する線が描画されます
  - ボーンを中心点をクリックするとボーンを選択することが出来ます
  - ボーンを中心点付近を移動するとボーン名が表示されます
- ボーンが「選択」に対応していない場合はボタンが無効になりクリックすることが出来ません
- ボタンを連続してクリックすると「選択」と「モードなし」で切り替えることが出来ます

ボーンは以下の条件に基づいて色がつけられます。また、移動可能なボーンは  $\delta\mathfrak{I}\mathfrak{U}$  で、回転可能なボーンは  $\delta\mathfrak{I}\mathfrak{S}\mathfrak{a}$  で表示されます。

条件	表示される色
選択されている	赤
変更してるがキーフレームに未登録	緑
IK リンク	黄
固定軸がある	紫
それ以外	青

### 3.7.2 回転モード（ボーン）

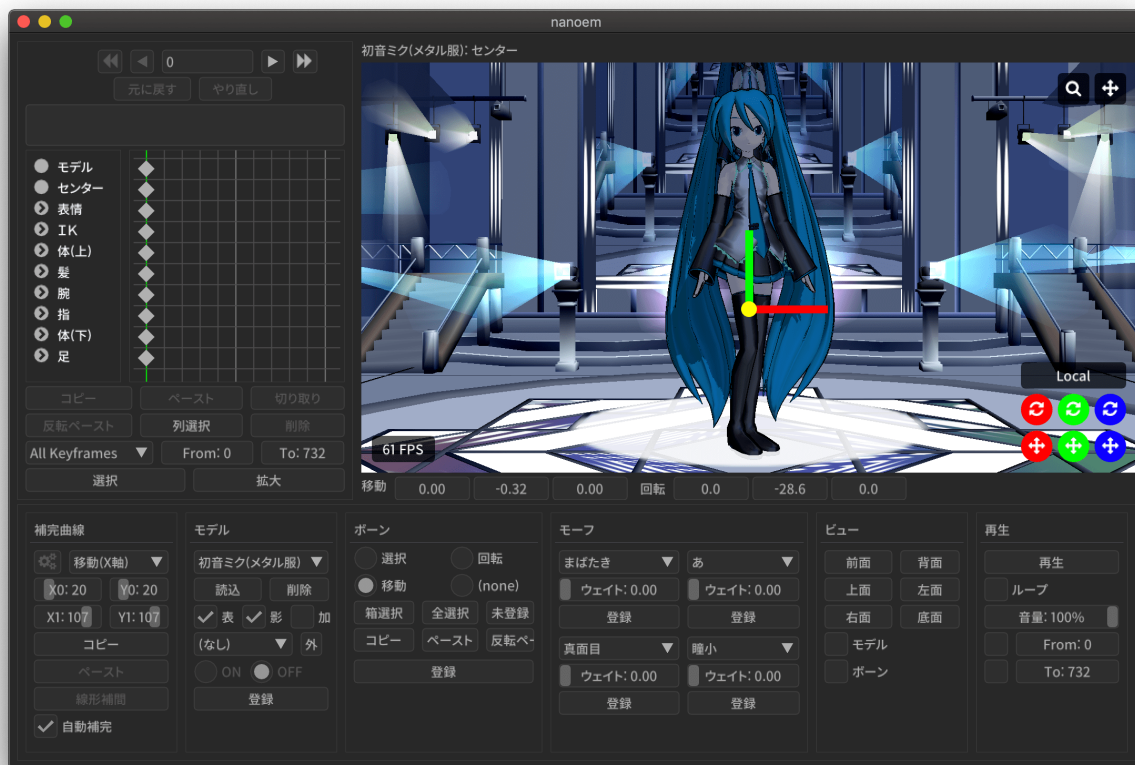


回転モードに切り替えます。選択してるボーンを中心に回転ハンドルが表示され、以下の条件に基づいて動作します。

ドラッグ箇所	結果
赤の部分	X 軸を中心に回転する
緑の部分	Y 軸を中心に回転する
青の部分	Z 軸を中心に回転する
黄の部分	何も反応しない

- ボーンが「回転」に対応していない場合はボタンが無効になりクリックすることが出来ません
- ボタンを連続してクリックすると「回転」と「モードなし」で切り替えることが出来ます
- 「登録」でキーフレームに登録されない限りフレーム切り替えで消失します
  - 「元に戻す」復元可能だが挙動の仕様上ややこしくなるため非推奨

### 3.7.3 移動モード（ボーン）



移動モードに切り替えます。選択してるボーンを中心に移動ハンドルが表示され、以下の条件に基づいて動作します。

ドラッグ箇所	結果
赤の部分	「カメラ視点からみて」X 軸固定で移動
緑の部分	「カメラ視点からみて」Y 軸固定で移動
黄の部分	ドラッグした箇所まで移動

- ボーンが「移動」に対応していない場合はボタンが無効になりクリックすることが出来ません
- ボタンを連続してクリックすると「移動」と「モードなし」で切り替えることが出来ます
- 「登録」でキーフレームに登録されない限りフレーム切り替えで消失します
  - 「元に戻す」復元可能だが挙動の仕様上ややこしくなるため非推奨

### 3.7.4 箱選択

箱選択モードに切り替えます。

- 一度クリックしてドラッグするとドラッグした範囲にあるボーンを全て選択することが出来ます
  - 主にボーンが多い手の部分を全て選択したい場合は「箱選択」を使ったほうがわかりやすく素早いですが
  - カメラの位置情報に依存するため、細かく選択したい場合はカメラをズームインしたほうがよいです
- 箱選択を解除したい場合は「回転」、「移動」、「全選択」、「未選択」をクリックすると解除出来ます

### 3.7.5 全選択

モデルにある全てのボーンを選択します。

---

注釈: 操作不可能なボーンは除外されます

---

### 3.7.6 未選択

ハンドルを用いて移動または回転したもののキーフレームに登録されていないボーンを全て選択します。

- 移動または回転させてかつキーフレームに登録されていないボーンは緑色表示になるのでそれらを全て選択する形です



### 3.7.7 コピー

選択されたボーンの移動量及び回転量をコピーします

---

注釈: タイムラインや [補完曲線パネル](#) にあるコピーとは独立しているためそれらと共有することは出来ません

---

### 3.7.8 ペースト

「コピー」されたボーンの移動量と回転量を現在のフレーム位置に対してペーストします

---

注釈:

- タイムラインや [補完曲線パネル](#) にあるコピーとは独立しているためそれらと共有することは出来ません
  - コピーされない限りボタンをおすことは出来ません
- 

### 3.7.9 反転ペースト

ボーン名の先頭に「右」または「左」が含まれている場合、それぞれ「左」または「右」の値になるように調整してペーストします

---

注釈:

- タイムラインや [補完曲線パネル](#) にあるコピーとは独立しているためそれらと共有することは出来ません
  - コピーされない限りボタンをおすことは出来ません
- 

### 3.7.10 物理

ご用心: 有効から無効に切り替える間は通過補完が発生するため、実際に無効にする数フレーム前に「物理演算有効」のボーンキーフレームを登録してください

ボーンキーフレームの登録時に物理演算を有効にするかどうかのチェックボックスです

- 物理演算が有効なボーンキーフレームはタイムライン上では  $\hat{N}$  マーカーで登録されます
-

- チェックボックスを外した状態で「登録」をするとボーンに紐づく剛体の物理演算が無効になった状態でボーンキーフレームに登録されます
  - ボーンに剛体が存在する場合にのみ有効です。存在しない場合は普通のボーンキーフレームの登録処理と同じです
- 物理演算が無効になったボーンキーフレームは  $\mathbf{a}i\mathbf{\check{N}}$  から  $\mathbf{\check{s}}\mathbf{\check{T}}\mathbf{\check{u}}$  に変化します
- 物理演算が無効になったボーンキーフレームを通過する際の剛体は「ボーン追従」扱いになります
- 物理演算の有効無効の状態はプロジェクト及び NMD に保存されます
  - VMD 自体の仕様で VMD には保存されません

### 3.7.11 登録

現在のフレーム位置に対してボーンのキーフレームに登録します。登録後移動または回転されたボーンの中心点の色が緑色から青色に戻ります

## 3.8 モーフパネル

注釈: モデルが選択されている場合のみ表示されます



モーフパネルはカテゴリごとに以下のように配置されます

左側	右側
目	リップ
まゆ	その他

### 3.8.1 (モーフ選択)

それぞれのカテゴリに属するモーフを選択します。

- 基本は一覧表示でそこから選択する形です
- 左右のボタンでモーフを切り替えることが出来ます

### 3.8.2 ウェイト

モーフのウェイト値を設定します。

- 基本は0で1.0にするとモーフの変形が行われます
- 空白部分をドラッグするとドラッグした分だけのウェイト値が設定されます
- 「登録」でキーフレームに登録されない限りフレーム切り替えで消失します
  - 「元に戻す」復元可能だが挙動の仕様上ややこしくなるため非推奨

### 3.8.3 登録

現在のフレーム位置に対してモーフのキーフレームに登録します

ご用心: 登録は「カテゴリごと」に行われるため、別のカテゴリでモーフのウェイトを変えた場合はそれぞれ「登録」を押す必要があります

## 3.9 アクセサリパネル



### 3.9.1 (アクセサリの選択)

---

注釈: 30.3.0 からアクセサリの名前変更機能が追加されました

---

一覧からアクセサリを選択します。

コントロールキーを押しながらクリックするとプロジェクト内のアクセサリの名前変更ができます（エンターキーを押すとアクセサリ選択に戻ります）。同じアクセサリを読み込ませた場合で名前を区別するときに利用してください。

### 3.9.2 読込（アクセサリ）

アクセサリ と同じです

### 3.9.3 削除（アクセサリ）

現在選択中のアクセサリを削除 と同じです

### 3.9.4 表示（アクセサリ）

アクセサリを表示するかを設定します。チェックを入れる则表示、外すと非表示になります。

### 3.9.5 外部親（アクセサリ）

アクセサリの接続先を設定します。

この機能は接続先モデルとそのボーンを設定することによりアクセサリの位置及び回転の計算を自動的に行なってくれます。これにより例えば接続先親モデルに装備品となるアクセサリをもたせるような動きで必要なアクセサリの位置及び回転の計算を肩代わりしてくれます。

同様の機能は [ボーンパネル](#) の「外（外部親設定）」がありますが、こちらはモデル専用です。

#### 接続先モデル

アクセサリが追従するモデルを設定します。デフォルトは地面（不動）です。

#### 接続先ボーン

アクセサリが追従するモデルのボーンを設定します。（なし）の場合は接続先モデルが「地面」で設定されているものと同じ扱いになります。

## 外部親登録

現在のフレーム位置に対して「接続先モデル」及び「接続先ボーン」を元にアクセサリのキーフレームを登録します。

### 3.9.6 移動

アクセサリの移動量を設定します

- クリックすると X/Y/Z 軸それぞれのパラメータが表示されます
  - 左右のボタンで 0.01 ずつ増減調整出来ます
  - 空白部分をドラッグすると移動した分のパラメータが増減します

ご用心: 「登録」でキーフレームに登録されない限りフレーム切り替えで消失します

### 3.9.7 回転

アクセサリの回転量を設定します

- クリックすると X/Y/Z 軸それぞれのパラメータが表示されます
  - 左右のボタンで 0.01 ずつ増減調整出来ます
  - 空白部分をドラッグすると移動した分のパラメータが増減します

ご用心: 「登録」でキーフレームに登録されない限りフレーム切り替えで消失します

### 3.9.8 Si

アクセサリの拡大率を設定します

ご用心: 「登録」でキーフレームに登録されない限りフレーム切り替えで消失します

### 3.9.9 Tr

アクセサリの不透明度を設定します

ご用心: 「登録」でキーフレームに登録されない限りフレーム切り替えで消失します

### 3.9.10 登録

現在のフレーム位置に対してアクセサリのキーフレームを登録します。

## 3.10 ビューパネル



#### 3.10.1 前面

カメラの角度を視点として前から見るように調整します

#### 3.10.2 背面

カメラの角度を視点として後ろから見るように調整します

#### 3.10.3 上面

カメラの角度を視点として上から見るように調整します

#### 3.10.4 左面

カメラの角度を視点として左から見るように調整します

#### 3.10.5 右面

カメラの角度を視点として右から見るように調整します

### 3.10.6 底面

カメラの角度を視点として下から見るように調整します

### 3.10.7 モデル 1.20 以降

現在選択中のモデル（厳密には特定のボーン）をカメラの目標点として調整し、かつ追従するようにします。これは以下の条件に従って処理されます

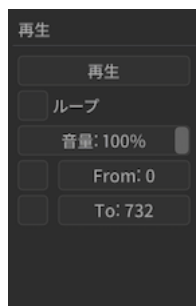
操作中心ボーンがある？	センター先の親ボーンがある？	選択されるボーン
あり	あり	操作中心ボーン
あり	なし	操作中心ボーン
なし	あり	センター先の親ボーン
なし	なし	モデル最初のボーン

操作中心ボーンは「標準ボーンプラグイン」で追加することが可能です。

### 3.10.8 ボーン 1.20 以降

現在選択中のボーンをカメラの目標点として調整し、かつ追従するようにします

## 3.11 再生パネル



### 3.11.1 再生

現在のプロジェクトを再生します

- 音源が設定されている場合は音源も同時に再生します

### 3.11.2 ループ

ループさせるかを設定します

- 「開始」と「終了」がそれぞれ設定されている場合その範囲内でループします
- デフォルトは無効なので終了まで達した場合はそのまま再生終了する

### 3.11.3 開始

「再生」における開始位置を設定します。

- 左にチェックボックスがあるためそれをチェックする必要がある
  - チェックしていない場合は開始位置設定にかかわらず 0 として認識される
  - デフォルトは無効
- タイムラインにある「開始」とは別

### 3.11.4 終了

「再生」における終了位置を設定します。

- 左にチェックボックスがあるためそれをチェックする必要がある
  - チェックしていない場合は終了位置設定にかかわらず全てのモーションの終端として認識される
  - デフォルトは無効
- タイムラインにある「終了」とは別

### 3.11.5 音量

---

注釈: 音量は音源が読み込まれないとスライダーが動かすことが出来ません

---

音源の音量を 0.0 から 1.0 の範囲内で設定します。



## 3.12 その他

### 3.12.1 クラッシュからのプロジェクトリカバリ

nanoem がクラッシュしたあとに起動するとリカバリ確認の画面が表示されます。OK を押すとクラッシュ直前までの操作内容にもとづいてリカバリを行います。

仕組みとして「ヘルプ」に *Open Redo Log Directory* が追加されていますが、そこで開いたフォルダに操作内容を逐次追記し、クラッシュ時にそれを読み込むことで実現しています。プロジェクトを作る度に作成されるので、不要であれば随時削除してください。

危険: 読み込んだアクセサリ、モデル、モーションの元となるファイルを移動させると正しく復元することが出来なくなるので一度読み込んだファイルを移動させないようにしてください

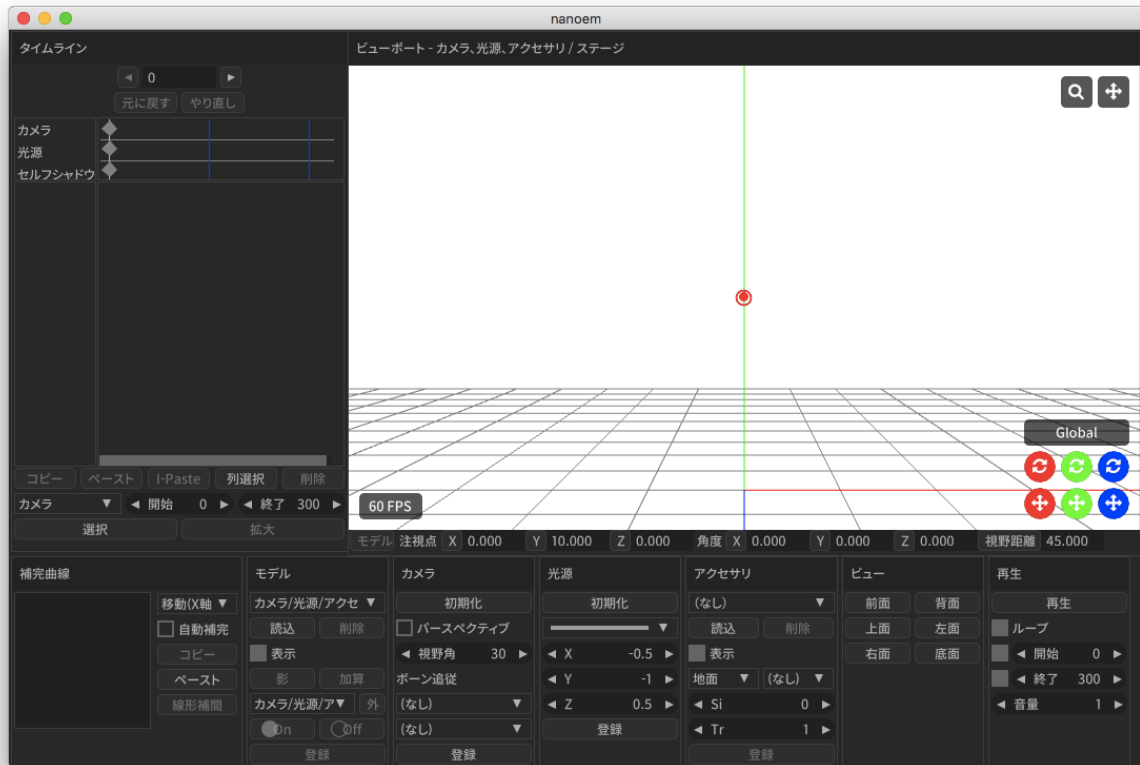
以下の操作内容がリカバリ可能です。

- アクセサリの読み込み
- モデルの読み込み
- モーションの読み込み
- キーフレーム登録及び削除
  - アクセサリ
  - ボーン
  - モーフ
  - カメラ
  - 光源
- ボーン変形
- モーフ変形

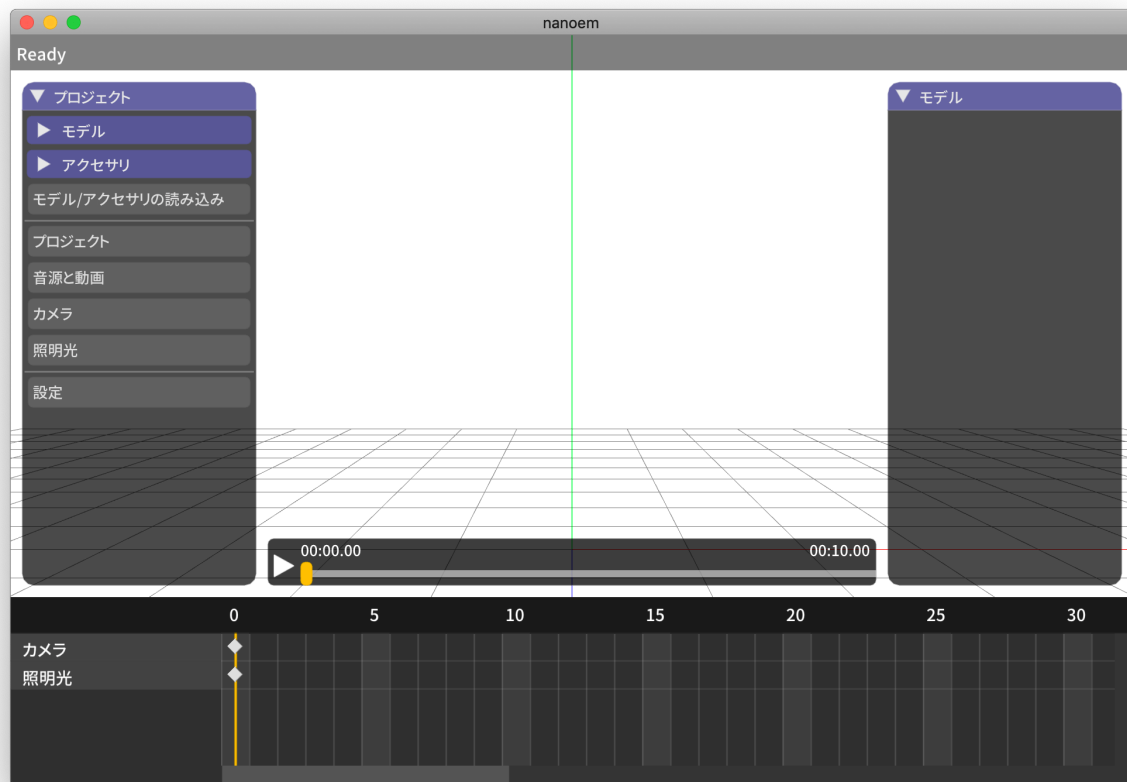
上記以外の、たとえばアクセサリの透明度や拡大率の設定は対象に入っていないので一回キーフレームに登録してください。

## 3.13 画面変遷の履歴

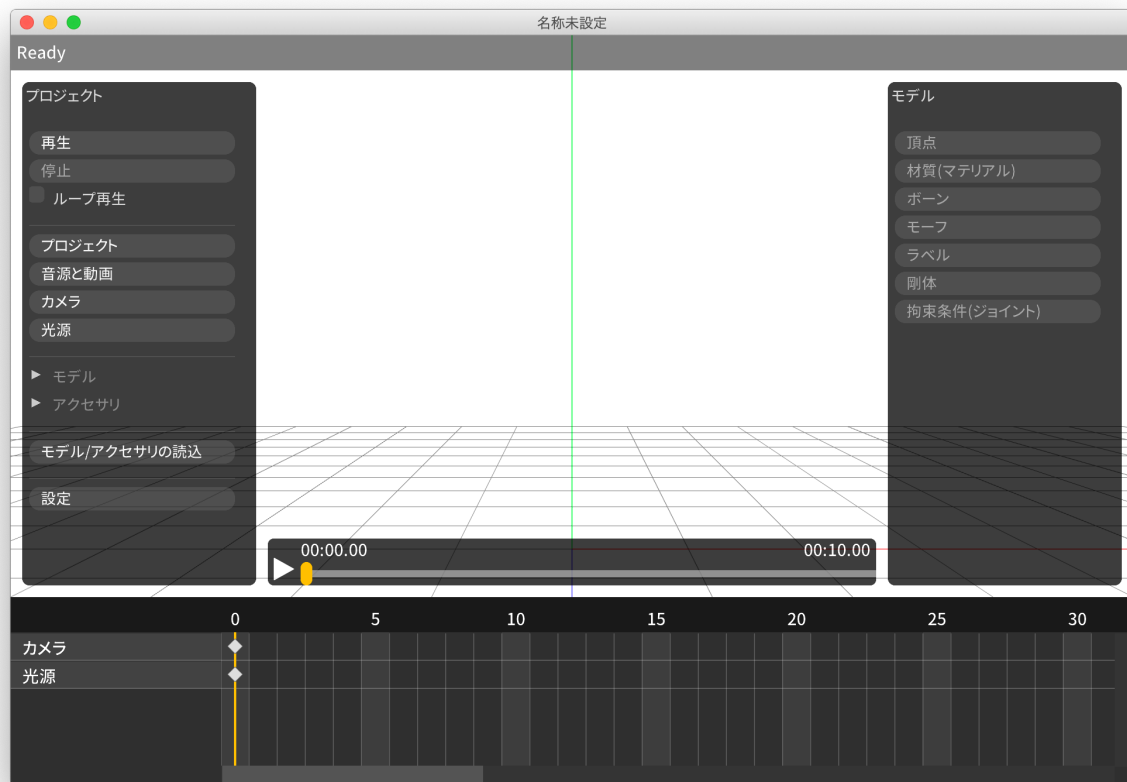
nanoem 1.15 から nanoem 23.x までは UI を全面的に変更し、MikuMikuDance (具体的には 9.26) と出来るだけ似せるように変えてました。



nanoem 1.14



nanoem 1.5



## 第 4 章

# アプリケーションのメニュー

- **Alt** がついてるものは巻き戻し処理が可能であることを示す

### 4.1 ファイル

#### 4.1.1 新規プロジェクト

プロジェクトを新規に作成します。

すでに何らかの変更が行われている場合は保存するかどうかの確認ダイアログが表示されます。新規プロジェクトが行われたあとは元のプロジェクトに巻き戻すことができなくなります。

#### 4.1.2 新規モデル

---

注釈: モデル編集用途の機能のためデフォルトでは表示されません。利用する場合は「[アプリケーションの設定](#)」から「[モデル編集機能を有効にする](#)」に **Alt** を入れる必要があります。

処理の特性上「[新規プロジェクト](#)」からプロジェクトを作成した場合は一回ファイルに保存してから実行する必要があります。

---

空のモデルを新規に作成してファイルに保存したうえで追加します。作成されるモデルデータは以下の仕様に基づきます。

- 「センター」ボーンのみ作成される
- 表示枠に「センター」ボーンの特権枠、及び空要素の「表情」表示枠が作成される
- 頂点、面、材質、モーフ、剛体、ジョイント、ソフトボディはそれぞれ空要素

モデル名は「新規モデル」にランダム生成の文字列の接尾詞がつく形で自動的に作成されます。

### 4.1.3 プロジェクトを開く

以下の形式のプロジェクトファイルを読み込みます。

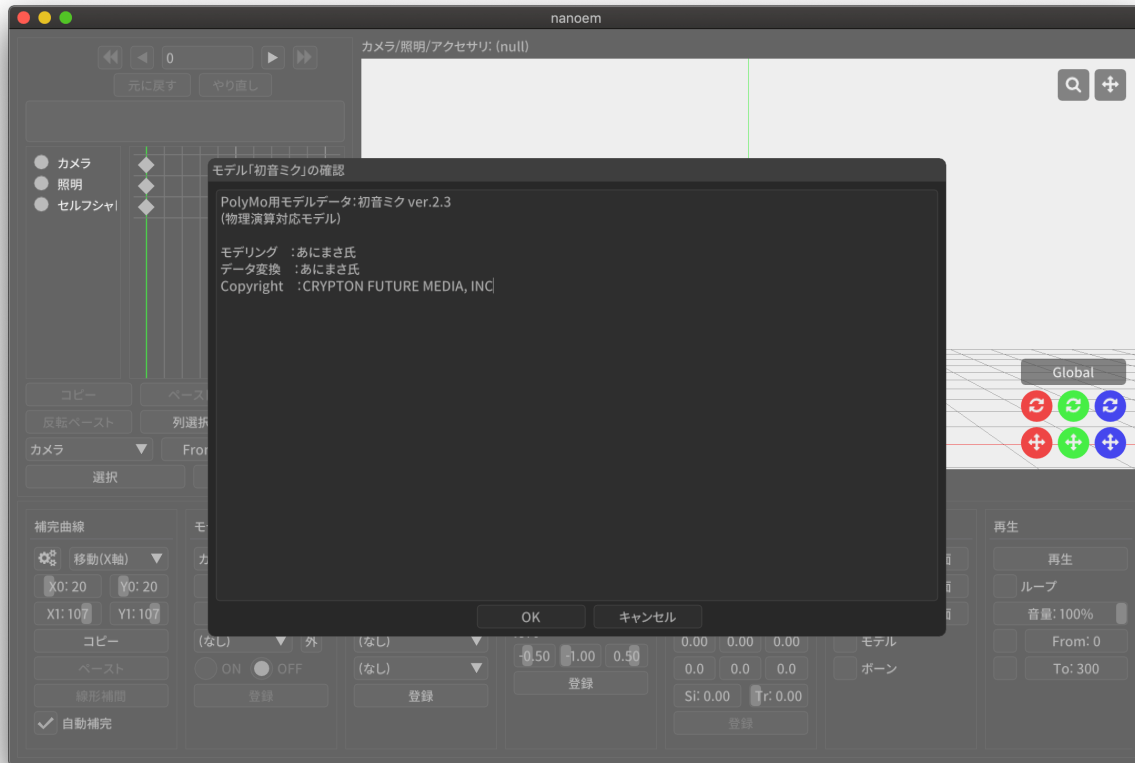
- .pmm
  - MikuMikuDance におけるプロジェクトファイル
  - 読み込み元のファイル移動させると読み込めなくなる
    - \* 読み込み元のファイルはモデル、アクセサリ、エフェクト、音源、背景動画が該当します
- .nmm
  - 1.23.0.0 から導入された第三世代のプロジェクトファイル
  - nma 形式と比較して保存は軽いが pmm 形式と同じく読み込み元のファイル移動させると読み込めなくなる
- .nma
  - nanoem における第二世代のプロジェクトファイル
  - 中身は zip のため拡張子を zip にして対応する解凍ソフトで解凍することが可能
  - 保存は重い（プロジェクト依存）が別の PC/Mac で読み込むことができる
    - \* pmm/nmm と違い中身が全て入ってるため読み込み元のファイルを削除しても復元できる
- .nanoem
  - nanoem における第一世代のプロジェクトファイル
  - 読み込み元の音源または背景動画を移動させると当該ファイルだけ読み込めなくなる
    - \* モデルとアクセサリとエフェクトは中に含まれてるため読み込み元を削除しても読み込める
  - macOS 版のみ

すでに何らかの変更が行われている場合は保存するかどうかの確認ダイアログが表示されます。プロジェクトファイルを開いて読み込まれた後は元のプロジェクトに巻き戻すことができなくなります。

### 4.1.4 開く

## モデル

.pmd または .pmx 形式のモデルファイルを読み込みます。同じモデルのファイル名に .fx がついてる場合はエフェクトも同時に読み込みます。



正常に読み込まれると確認ダイアログが開き OK をクリックすることで初めて表示されます。キャンセル をクリックすると以降の読み込みがキャンセルされます（失敗時はエラーダイアログが表示されます）

エフェクトファイルがあり、かつエフェクトプラグインが無効の場合は有効にするかの確認画面が表示されます。キャンセルした場合はモデル読み込みがキャンセルされます。

ちなみに：メニューから読み込む方法以外にも以下の方法で読み込むことができます（確認ダイアログが出る点は共通）

- ファイルを選択し nanoem のウィンドウにドラッグ・アンド・ドロップ
- 「モデル」パネルの「読み込む」からファイル選択

また、エフェクトファイルが同時に読み込まれている場合はエフェクトファイルに対して書き換えを行うと自動的にエフェクトをリロードします。

## アクセサリ

.x 形式のアクセサリファイルを読み込みます。同じアクセサリのファイル名に .fx がついてる場合はエフェクトも同時に読み込みます。

正常に読み込まれるとそのままアクセサリが表示され、「アクセサリ」パネルに読み込まれたアクセサリが追加されます（失敗時はエラーダイアログが表示されます）

エフェクトファイルがあり、かつエフェクトプラグインが無効の場合は有効にするかの確認画面が表示されます。キャンセルした場合はアクセサリ読み込みがキャンセルされます。

---

ちなみに：メニューから読み込む方法以外にも以下の方法で読み込むことができます。

- ファイルを選択し nanoem のウィンドウにドラッグ・アンド・ドロップ
- 「アクセサリ」パネルの「読み込む」からファイル選択

また、エフェクトファイルが同時に読み込まれている場合はエフェクトファイルに対して書き換えを行うと自動的にエフェクトをリロードします。

---

## モデルポーズ

ご用心:

- ポーズが読み込まれてもキーフレームは登録されないためシークすると元に戻ってしまうため必ずキーフレームを登録してください
- モデルが選択されていないと実行できません

.vpd 形式のポーズファイルを読み込みます。

正常に読み込まれるとポーズファイルの内容にあわせてモデルが変形します（失敗時はエラーダイアログが表示されます）。MikuMikuMoving が採用しているモーフ変形が追加された拡張形式にも対応しています。

## モーション

.vmd または .nmd 形式のモーションファイルを読み込みます。

## モデルモーション

---

重要:

---



- 33.4 からモーション読み込み後にモデルに含まれるボーン名またはモーフ名と一致するキーフレームが全て選択されるようになりました。
- 33.0 からモデルに存在しないボーンまたはモーフがキーフレームにある場合は存在しない対象名をダイアログで表示するようになりました。
- 33.0 からカメラモーションが読み込まれた場合はエラーを出すようになりました。それより前のバージョンでは単純に無視されます。

ご用心: モデルが選択されていない場合は実行できません。

モーションファイルを選択されたモデルに対して読み込みます。

選択されたモデルに存在するボーンまたはモーフに対して変形を行います。モーションに登録されてるボーンまたはモーフがモデルに存在しないキーフレームは無視されます。

## カメラモーション

重要: 33.0 からモデルモーションが読み込まれた場合はエラーを出すようになりました。それより前のバージョンでは単純に無視されます。

選択されたカメラモーションを読み込み、カメラに対してキーフレームに設定します。

## 光源モーション

重要: 33.0 からモデルモーションが読み込まれた場合はエラーを出すようになりました。それより前のバージョンでは単純に無視されます。

選択されたカメラモーションを読み込み、光源に対してキーフレームに設定します。

## 音源

音源をプロジェクトに読み込みます。

注釈: プロジェクトに読み込める音源はひとつまでです。別の音源を読み込んだ場合以前の音源を削除してから読

み込まれます。

---

読み込み可能な音源ファイルは nanoem 自体のバージョン及び中に含まれるコーデックによって異なるため、実際に読み込みできるかを必ず確認してください。

- Windows 版及び macOS 版共通で読み込み可能
  - リニア PCM (いわゆる非圧縮の音源)
  - MP3
- macOS 版で読み込み可能
  - AAC

## 背景動画

動画をプロジェクトに読み込みます。読み込まれた動画はビューポートの背景動画として表示されます。

---

注釈: プロジェクトに読み込める動画はひとつまでです。別の動画を読み込んだ場合以前の動画を削除してから読み込まれます。

---

---

ちなみに: 動画の位置または拡大率は画面右上のカメラズーム及び矢印アイコンを SHIFT キーを押しながら操作すると可能です。

---

読み込み可能な動画ファイルは nanoem 自体のバージョン及び中に含まれるコーデックによって異なるため、実際に読み込みできるかを必ず確認してください。

- Windows 版及び macOS 版共通で読み込み可能
    - RawVideo (いわゆる非圧縮の動画)
    - [UtVideo Codec](#)
    - [Grass Valley HQX Codec](#)
- \* [UtVideo](#) 及び [Grass Valley HQX](#) は圧縮動画形式であり展開するためのデコード負荷が大きいため動画によっては FPS を下げる要因になることがあります

## 4.1.5 保存

プロジェクトファイルを書き出します。

新規プロジェクトから作成した場合は初回はファイルダイアログが開きます。また、二回目以降あるいは「プロジェクトを開く」から開いた場合は同じファイルに対して保存します。

- Windows では *.nma* または *.nmd* または *.pmm* 形式
- macOS では *.nma* または *.nmd* または *.pmm* または *.nanoem* 形式

#### NMA/NMM 形式の保存の制限について

- nma/nanoem 形式から nmm/pmm 形式への保存はできません
- nmm 形式から nma/nmd/nanoem 形式に保存することは可能です
- nma 形式から nmm 形式に保存する場合は以下の手順を実施することで可能です
  - nma ファイルを拡張子 zip にリネームして解凍する
  - 新規プロジェクトでプロジェクトをリセットする
  - 解凍して出てきたファイルを全て読み込む
  - nmm 形式で保存する

保存元と保存先のマトリックスは以下のとおりです（下方向は保存元、右方向は保存先を示す）

	nanoem	nma	nmm	pmm
nanoem	⚠	⚠	⚠	⚠
nma	⚠	⚠	⚠	⚠
nmm	⚠	⚠	⚠	⚠
pmm	⚠	⚠	⚠	⚠

#### PMM 形式の保存の制限について

**重要:** pmm 形式に書き出すときは元のプロジェクトファイルを保持してください。

MMD に読み込ませる目的で保存する場合は MMD の実行ファイルと同じ階層にある UserFile フォルダに保存する必要があります。これは MMD の仕様として UserFile を基準にファイルが読み込まれるためです。

31.0 から pmm 形式の保存が実装されました。ただし仕様の違いから以下の注意点があります。

- MMD 9.0 以降で読み込み可能な PMM 2.0 形式で保存されます
  - MMD 9.0 未満で読み込み可能な PMM 1.0 形式では保存されませんのでご注意ください
  - PMM 1.0 形式への保存の対応予定はありません

- モデル及びアクセサリは両方あわせて最大 255 体までです
- モデルまたはアクセサリのパス名は 255 bytes までです。これより長いと読み込めません
  - 日本語のファイル名を用いた場合は 255 bytes より短くなります
  - ShiftJIS で保存されるためパス名に絵文字などを使うと読み込めなくなります
- 描画順の仕様の違いにより書き出し時にすべてのモデル、すべてのアクセサリの順番で保存します
  - 例えば「モデル、アクセサリ、モデル」の描画順の場合は「モデル、モデル、アクセサリ」に整理されます
- MMD の仕様で読み込み可能なキーフレーム数に上限があります
  - カメラは 1 万キーフレーム、モデルは 5 万キーフレームを上限とする制約があるようです
  - 大量のキーフレームが登録される「[全てのモデルのモーションを焼き込む \*âĤĤ\*](#)」を実施するとキーフレーム数制約が引かかる可能性があります
- 表示枠の開閉状態は保存されず、全て閉じた状態で保存されます
- 重力キーフレームは nanoem では未実装のため保存されません
  - キーフレームのみで重力設定自体は保存されます
- レイアウト情報は保存されません
- (33.4 未満の場合のみ) キーフレームの選択状態が保存されません

#### 4.1.6 名前をつけて保存

プロジェクトの変更状態に関係なく常にファイルダイアログが開いて保存先を指定する必要があることを除いて「保存」と同じです。

#### 4.1.7 書き出し

モデルポーズ

ご用心:

- モデルポーズはモーションと異なりキーフレームではなく「全てのボーン及びモーフの状態」を保存します
  - 上記の仕様によりキーフレーム登録しなくても変形状態が保存されます

現在のフレーム位置から選択された全てのボーン及びモーフの状態を `.vpd` 形式としてファイルに書き出します。

## モデルモーション

ご用心:

- モデルが選択されていないと実行できません
- モーション読み込み時にボーンまたはモーフがモデルに存在せずに無視されたキーフレームは登録されません

選択されたモデルにある全てのキーフレームを `.vmd` または `.nmd` 形式としてファイルに書き出します。

## カメラモーション

カメラの全てのキーフレームを `.vmd` または `.nmd` 形式としてファイルに書き出します。

## 光源モーション

光源の全てのキーフレームを `.vmd` または `.nmd` 形式としてファイルに書き出します。

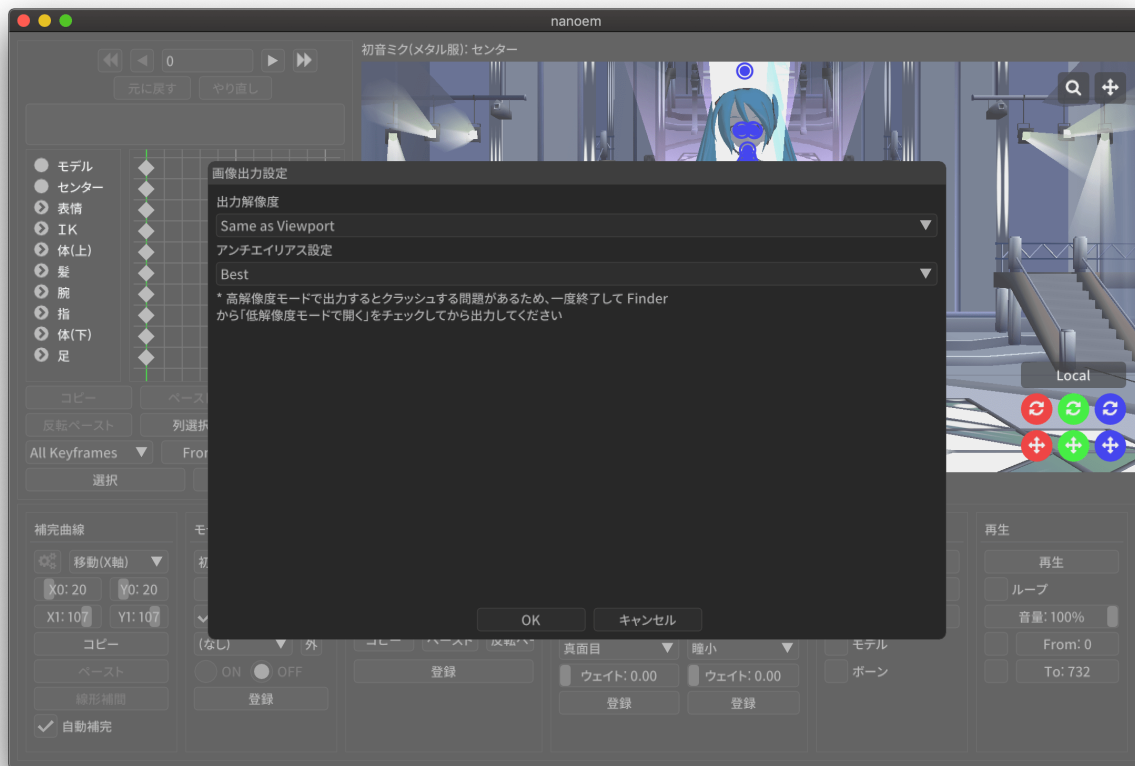
## モデル 1.20 以降

現在選択中のモデルを `.pmx` 形式としてファイルに書き出します。

ご用心:

- nanoem 起動時に設定項目の「モデル編集機能を有効にする」が `OFF` されていないと表示されません。そのため、当該項目を `ON` したあとに nanoem を再起動する必要があります
- 書き出し先は読み込み時と同じフォルダに指定してください。モデルフォーマットの仕様として違うフォルダで書き出すとテクスチャが正しく表示されなくなります

## 画像（書き出し）



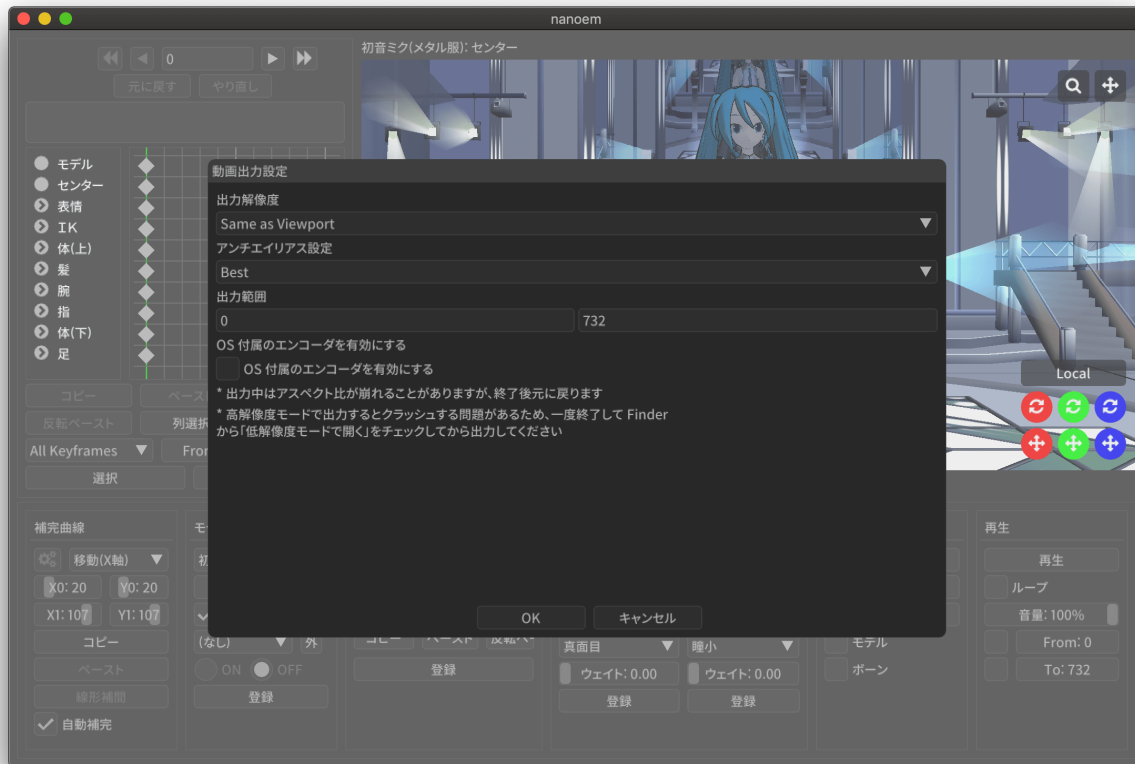
### ヒント:

- 透過を含める場合は `.png` 形式で出力してください。それ以外の形式だと透過を含むことができない仕様のため正しく表示されないことがあります。
- 画像出力は一回物理演算をリセットしてフレーム移動を行ってから書き出しを行う仕組みとなっています。この仕様により、そのまま画像出力を行うと物理演算を再現出来ない問題があります。
  - 物理演算を維持して書き出したい場合は「**全てのモデルのモーションを焼き込む *ARt***」を行って画像出力用のプロジェクトとして書き出した後に画像出力を行ってください。

現在のフレーム位置のビューポートを画像として書き出します。以下の手順により書き出しが始まります。

1. プロジェクトが未保存の場合プロジェクト保存の確認画面が出ます
2. 最初に書き出し設定（サイズ指定など）のダイアログが出ます
3. 保存先を指定するファイルダイアログが開きます
4. 保存先に画像書き出しが始まります

## 動画（書き出し）



動画としてファイルに書き出します。以下の手順により書き出しが始まります

## ご用心:

- 「OS 付属のエンコーダを有効にする」はほぼリアルタイムな書き出しが可能ですが、有効にして書き出した場合「動画出力で書き出すと再生では起こらなかったフレーム欠けが起こる」の問題が発生します。完全な動画として出力したい場合は無効にしてください。
- 出力される動画のフレームレートは「再生時フレームレート」に依存し 30/60 FPS です。ドロップフレームはありません。
  - 例えばドロップフレームを入りたい、あるいは 24 FPS に落としたい場合は「[動画編集できますか？](#)」に紹介されている動画編集ソフトを利用して変更してください。

## ヒント:

- 透過を含める場合は「[透過つき動画を出力したい](#)」を参照してください。

- 動画出力は画像出力と同じように一回物理演算をリセットしてフレーム移動を行ってから書き出しを行う仕組みとなっています。これはフレームの範囲指定をした場合に物理演算を再現出来ない問題があります。
    - そのため、物理演算を維持して書き出したい場合は「**全てのモデルのモーションを焼き込む *àRI***」を行って動画出力用のプロジェクトとして書き出した後に動画出力を行ってください。
- 

1. プロジェクトが未保存の場合プロジェクト保存の確認画面が出ます
  2. 動画設定のダイアログが開きます
  3. (「OS 付属のエンコーダを有効にする」の設定によって変化します)
    - 「OS 付属のエンコーダを有効にする」にチェックした場合は4に進みます
    - 「OS 付属のエンコーダを有効にする」にチェックを外した場合は Plugins からプラグインを指定します
      - どれを使うか迷う場合は `plugin_lsmash` を推奨します
  4. 保存先を指定するファイルダイアログが開きます
  5. 保存先に動画書き出しが始まります
- 

注釈: 画像書き出しと違って動画書き出しは時間がかかりかつ負荷のかかる作業のため、書き出し前は一旦 nanoem 以外のアプリケーションを全て終了させ、書き出し中は他のアプリケーションを立ち上げたりせずに待ちましょう。

---

## 動画設定

- 出力解像度
    - 動画出力するときの解像度を以下から選択します。
      - \* ビューポートと同じ
      - \* Application Default (960x640)
      - \* XGA (1024x768)
      - \* 720p (1280x720)
      - \* WXGA (1600x900)
      - \* UXGA (1600x1200)
      - \* 1080p (1920x1080)
      - \* WUXGA (1920x1200)
-



- \* WQHD (2560x1440)
- \* WQXGA (2560x1600)
- \* 4K (3840x2160)
- \* 8K (7680x4320)
- 「ビューポートと同じ」は「ビューポート設定」の「出力サイズ」の設定が使われます。
- 解像度が高ければ高いほど出力に時間がかかります。
- ビューポートのアスペクト比を維持
  - 有効にするとビューポートのアスペクト比を利用して出力します
    - \* これにより無効時の出力解像度からアスペクト比を決定することによる歪みを防ぐことができます
    - \* また縦長解像度（出力解像度を選択肢がないためビューポート設定の必要あり）の出力を正しく行うことができるようになります
  - 無効にすると 26.0 より前と同じ挙動で出力解像度からアスペクト比が決定されます
- フレームズレ抑止を有効にする
  - 垂直同期を上回るフレームレートを出すとフレーム移動の際にズレが生じることがあるため、これを抑止します
  - 無効にすると垂直同期が無効になるため書き出し速度が早くなりますが、先のフレームズレの問題が生じるためあくまで製作途中の確認用として使ってください
  - 「OS 付属のエンコーダを有効にする」を有効にしている場合は影響を受けません
- アンチエイリアス設定
  - [アンチエイリアス設定](#) と同じですが、動画出力するときのみ適用されます。
  - 頻繁に出力する必要がある場合は無効に、動画書き出しの最終出力するとき最高に設定することを推奨します。
  - エフェクト（主にポストエフェクトが対象）によってはアンチエイリアスを明示的に無効にする必要があります。
    - \* エフェクトの取扱説明書にその必要があるかどうか記載されているので利用時に確認してください。
- 出力範囲
  - 動画を出力する範囲を設定します
  - デフォルトはプロジェクトの最初から最後までです

- Plugins

- 動画書き出しに利用するプラグインを指定します

- \* plugin\_avfoundation (macOS 版のみ)

- ・ AVFoundation を使った動画出力をするプラグインです
      - ・ 保存可能な形式は mp4/mov です
      - ・ 設定可能な項目はありません

- \* plugin\_ffmpeg

- ・ ffmpeg の内部ライブラリを使った動画出力するプラグインです
      - ・ 保存可能な形式は avi のみです
      - ・ Video Codec
      - ・ 保存する動画形式で以下から選択します
      - ・ Raw Video
      - ・ 無圧縮動画で plugin\_lsmash と同一です
      - ・ UT Codec Video
      - ・ 可逆圧縮動画で圧縮処理のために保存に時間がかかるかわりにサイズを概ね半分にできます
      - ・ Video Pixel Format
      - ・ 動画出力で格納する画像の保存形式で以下から選択します
      - ・ RGB
      - ・ RGBA
      - ・ YUV420P
      - ・ YUV422P
      - ・ YUV444P
      - ・ 基本は RGB で、画像劣化を許容してでもサイズを小さくしたい場合は YUV を使います
      - ・ Audio Codec
      - ・ 現時点で PCM のみです

- \* plugin\_lsmash

- ・ 無圧縮動画を出力するプラグインです

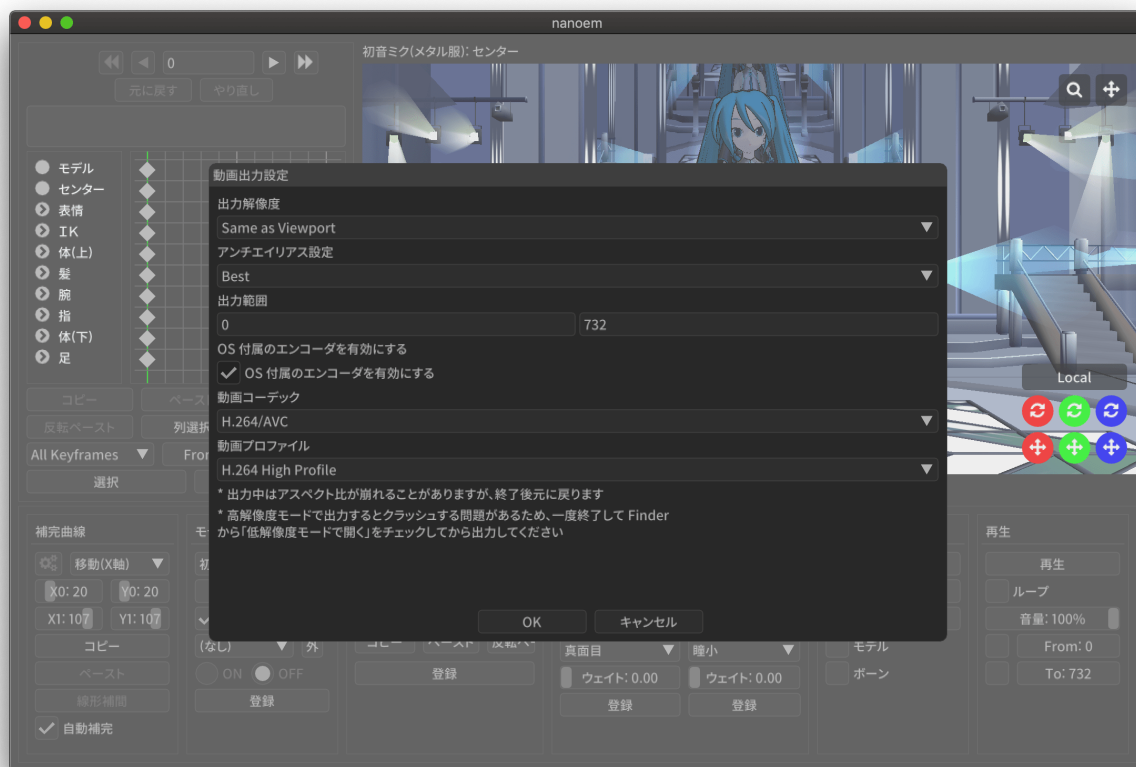
- ・ AviUtl のプラグインのひとつである lsmash-works とは別物なのでご注意ください
- ・ 保存可能な形式は mov のみです
- ・ Estimated export video size は動画出力後の見積もりサイズです
- ・ 例えば 3 分間の 1280x720 の 60FPS の動画を出力すると約 30GB になります

\* plugin\_gif

- ・ gif を出力するプラグインです
- ・ 保存可能な形式は gif のみです
- ・ 設定可能な項目はありません
- ・ エンコード処理の関係上、動画出力完了後すぐに画面が戻りませんが時間経過により復帰します
- ・ 将来的に廃止される可能性があるため利用非推奨です
- ・ 代替手段として mp4 形式で動画出力し、[gifski](#) を利用してエンコードすることを推奨します

v1.18 から「OS 付属のエンコーダを有効にする」が追加されました。以下のエンコード設定が可能になり、かつ高速に処理出来ることが特徴です。

ただし「よくある質問と回答」にあるようにフレームが欠けることがあるため完全な動画で保存する場合はプラグイン利用の動画出力を推奨します。



- 音声コーデック
  - 動画出力時に音源が読み込まれてる場合に選択可能です
  - デフォルトは AAC のみですが、動画出力時のファイルの拡張子を *mov* にすると *Apple Lossless* と *Linear PCM* が追加で選択出来ます
- 動画コーデック
  - デフォルトで *H.264/AVC* のみですが、動画出力時のファイルの拡張子を *mov* にすると *Apple ProRes 422* と *Apple ProRes 4444* が追加で選択出来ます
- 動画のピクセル形式
  - 現時点で *RGBA 8bits* のみ選択可能です
  - 今後の HDR 対応により *RGB 10bits A 2bits* と *RGBA 16bits float* の追加予定があります
- 動画プロファイル
  - 動画プロファイルは動画コーデックに *H.264/AVC* を選択した場合のプロファイルを設定します。それ以外のコーデックでは無視されます。

### 4.1.8 終了

アプリケーションを終了させます。

プロジェクトに未保存の変更があった場合終了確認の画面が出ます。

## 4.2 編集

### 4.2.1 元に戻す

重要: 27.0.0 よりプロジェクト単位のみからプロジェクト単位（カメラモード時）とモデル単位（モデル選択時）で分離されました。これにより以下の変更があります。

- モデル操作はモデル単位で独立して「元に戻す」が可能になります
- モデル単体で完結しない操作（例えばフレーム挿入または削除）はプロジェクト単位で記録されるため一回カメラモードに切り替える必要があります
- アクセサリ操作はカメラモードから操作する関係でプロジェクト単位で記録されます

前回の操作を巻き戻します。

- ボーン操作などの操作中にフレーム移動が発生した場合は移動前に実施した処理が「元に戻す」の起点となります
- 「元に戻す」及び「やり直し」の操作は 64 回まで保存されます。それ以上前の操作は消去されます
- マウス側に「元に戻す」ボタンがあってそれを押された場合は「元に戻す」の処理が実行されます

### 4.2.2 やり直し

ご用心:

- 一回「元に戻す」で別の操作を行った場合はやり直しができなくなります
  - 例として  $A \rightarrow B \rightarrow [\text{元に戻す}] \rightarrow A \rightarrow C$  とした場合  $B$  の操作は失われる

「元に戻す」で戻した操作を取り消します。

- 「元に戻す」及び「やり直し」の操作は 64 回まで保存されます。それ以上前の操作は消去されます。

- ・ マウス側に「やり直し」ボタンがあってそれを押された場合は「やり直し」の処理が実行されます

### 4.2.3 切り取り $\hat{A}Rt$

選択された全てのキーフレームをコピーした上で削除します。

### 4.2.4 コピー

モデルが選択されててかつボーンパネルが「選択」状態の場合は選択されたボーンの移動及び回転の状態をコピーします。

それ以外の場合は選択された全てのキーフレームをコピーします。

### 4.2.5 貼り付け $\hat{A}Rt$

モデルが選択されててかつボーンパネルが「選択」状態の場合は選択されたボーンの移動及び回転の状態を上書きする形で貼り付けします。

それ以外の場合は「コピー」あるいは「元に戻す」で選択されたキーフレームを現在のフレーム位置を基準にペースト（貼り付け）します。

---

注釈: キーフレームが重なる場合はコピーされたキーフレームが優先され、登録されているキーフレームは上書きされます。

---

### 4.2.6 全てのキーフレームを選択

全てのキーフレームを選択します。

カメラモードの場合はカメラ、光源に登録されている全てのキーフレームを選択します。

モデルモード（モデルが選択されている）の場合はモデル、ボーン、モーフに登録されている全てのキーフレームを選択します。

### 4.2.7 モーション

#### 空フレームを挿入 $\hat{A}Rt$

現在のフレームを基準に基準より後ろに登録されているキーフレームを全てひとつ後にずらします。

対象となるモーションは以下の上から順番に処理されます。

前提条件	対象
モデルが選択されている場合	選択されたモデルのモーション
モデルが選択されていない場合	カメラ/照明/セルフシャドウ 及び全てのアクセサリのモーション

#### 現在のフレームを削除 **⌘Rt**

現在のフレームを基準に基準より後ろに登録されているキーフレームを全てひとつ前にずらします。ずらした先にキーフレームが登録されている場合そのキーフレームは削除されます。

対象となるモーションは「**空フレームを挿入 ⌘Rt**」に従います。

#### 初期化 **⌘Rt**

選択されたモーションの全てのキーフレームを削除し、最初にあるキーフレームを初期化します。

初期化対象となるモーションは以下の上から順番に処理されます。

前提条件	対象
モデルが選択されている場合	選択されたモデルのモーション
アクセサリが選択されている場合	選択されたアクセサリのモーション
上記のどちらも当てはまらない場合	カメラ/照明/セルフシャドウ のモーション

### 4.2.8 ボーン

#### 数値入力ダイアログを開く **⌘Rt**

選択されたボーンに対する数値入力のダイアログを開きます。

ビューポート下にある数値入力と同じです。

#### 補正ダイアログを開く **⌘Rt**

---

注釈: あらかじめ補正対象となるボーンキーフレームを選択する必要があります。何も選択されていない場合は一切補正されません。

---

補正ダイアログを開きます。

選択された全てのボーンのキーフレームの移動、回転を対象に乗算及び加算を用いて数値を調整します。主に大きさが異なるモデルのためにモーション全体の動きを一括で補正するときに使います。

バイアス指定ダイアログを開く

バイアス指定ダイアログを開きます。

選択モード

ボーンを選択するための選択モードに切り替えます。

回転モード

ボーンを回転させるための回転モードに切り替えます。

移動モード

ボーンを移動させるための回転モードに切り替えます。

角度をリセット **âRt**

選択されたボーンの X/Y/Z 軸の角度をすべて 0 にリセットします。

## 4.2.9 カメラ

数値入力ダイアログを開く **âRt**

カメラに対する数値入力のダイアログを開きます。

ビューポート下にある数値入力と同じです。

補正ダイアログを開く **âRt**

---

注釈: あらかじめ補正対象となるカメラキーフレームを選択する必要があります。何も選択されていない場合は一切補正されません。

---

補正ダイアログを開きます。

選択された全てのカメラのキーフレームの注視点、角度、視野距離を対象に乗算及び加算を用いて数値を調整します。



#### 角度をリセット **âRt**

カメラの X/Y/Z 軸の角度を 0 にリセットします。

注視点や視野距離も一緒にリセットしたい場合はカメラパネルの「初期化」を使用してください。

### 4.2.10 モーフ

#### 補正ダイアログを開く **âRt**

補正ダイアログを開きます。

選択された全てのモーフのキーフレームのウェイトを対象に乗算及び加算を用いて数値を調整します。

#### 唇のキーフレームを削除 **âRt**

唇カテゴリの全てのキーフレームを選択状態にかかわらず削除します。

#### 目のキーフレームを削除 **âRt**

目カテゴリの全てのキーフレームを選択状態にかかわらず削除します。

#### まゆのキーフレームを削除 **âRt**

まゆカテゴリの全てのキーフレームを選択状態にかかわらず削除します。

#### 全てのモーフをリセット **âRt**

全てのモーフのウェイトを選択状態にかかわらず 0 にリセットします。

#### 全てのキーフレームを登録 **âRt**

現在のフレーム位置にある全てのモーフをキーフレームに登録します。

すでに登録されている場合は無視されます。

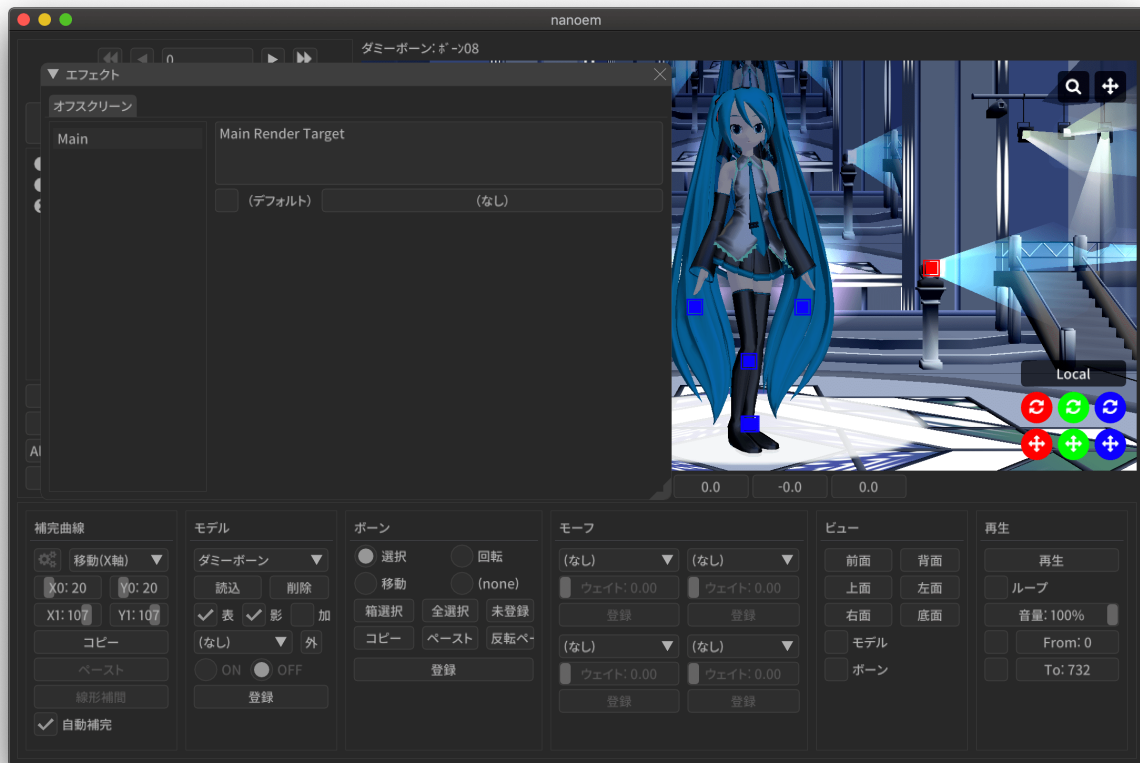
### 4.2.11 エフェクト設定ウィンドウを開く

---

**重要:** エフェクトファイル単体のドラッグ&ドロップはエフェクト設定ウィンドウが開いている時のみ利用可能です。開いていない時にドラッグ&ドロップをするとエラーが表示されます。

---

エフェクト設定ウィンドウを開きます。



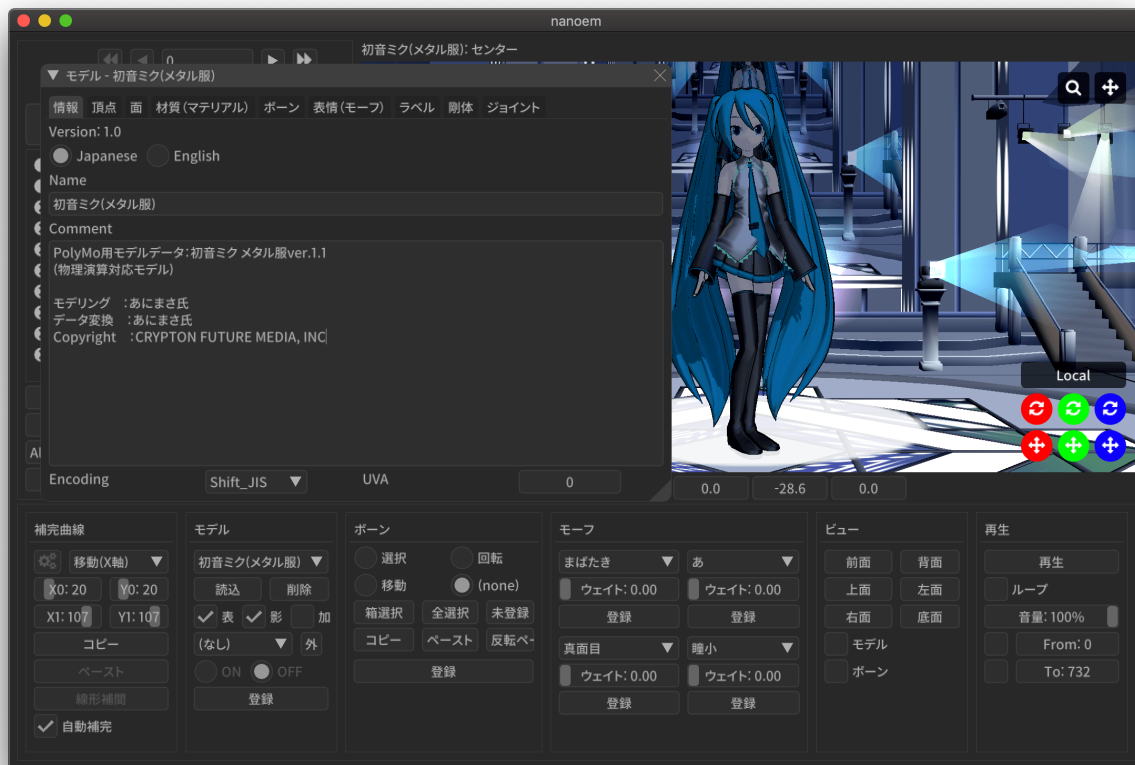
詳細は「エフェクトの設定」を参照してください。

#### 4.2.12 モデル編集ウィンドウを開く

ご用心:

- nanoem 起動時に設定項目の「モデル編集機能を有効にする」が ☐ されてないと表示されません。
- この機能は今後仕様が変化する可能性があります

モデル編集ウィンドウを開きます。詳細は model を参照してください。モデルが選択されていない場合は実行できません。



#### 4.2.13 モデルプラグイン

モデルプラグインを実行します。モデルが選択されていない場合は実行できません。

利用可能なプラグインは [プラグイン](#) を参照してください。

#### 4.2.14 モーションプラグイン

モーションプラグインを実行します。モデルが選択されている場合はモデルのモーションに、モデルが選択されていない場合はカメラ、照明、アクセサリのモーションに対して適用します。

利用可能なプラグインは [プラグイン](#) を参照してください。

## 4.2.15 設定

ご用心: macOS 版ではこの項目は出ず、 nanoem > 設定... が該当します。

設定ウィンドウを開きます。詳細は [アプリケーションの設定](#) を参照

## 4.3 プロジェクト

### 4.3.1 再生

現在のフレーム位置からプロジェクトを再生します。

### 4.3.2 停止 (再生中の場合は「再開」)

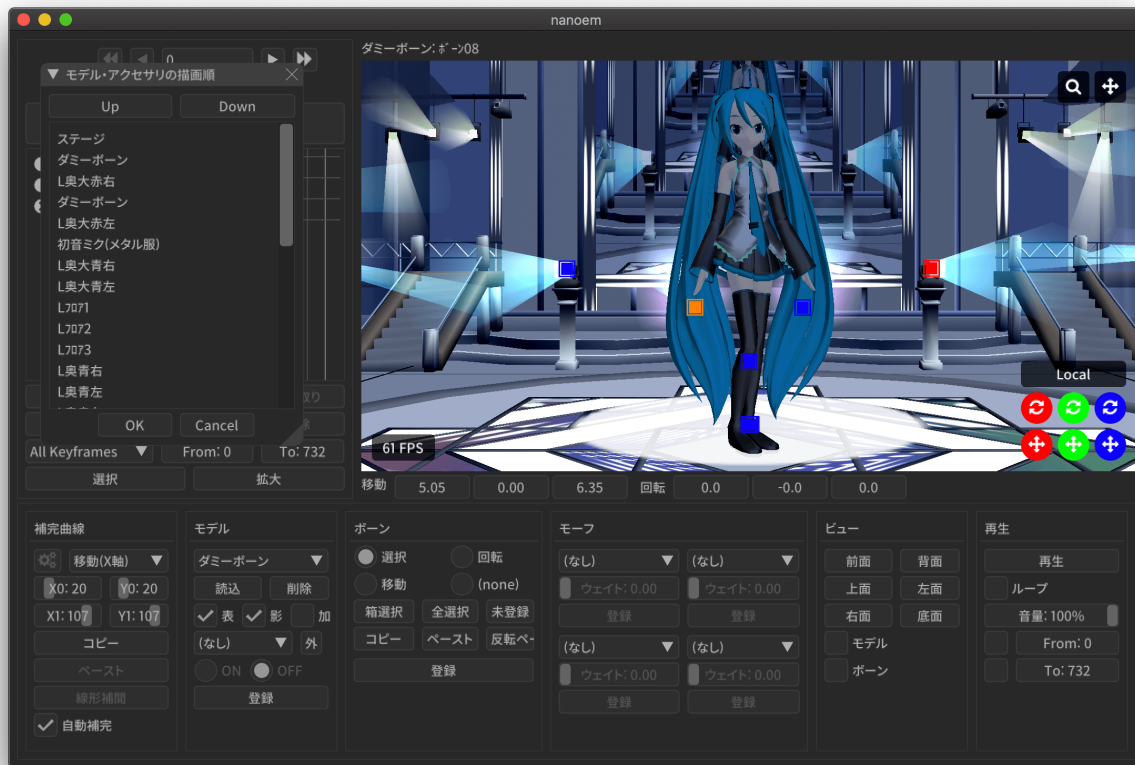
再生中のプロジェクトを停止します。再生中の場合は現在位置から再生を再開します。

### 4.3.3 ビューポート設定ダイアログを開く

ビューポートの設定を変更するためのダイアログを開きます。

- 出力サイズ (幅 / 高さ)
  - ビューポートに表示する大きさの既定値を設定します
  - ウィンドウサイズの大きさにあわせて計算されるため実際に表示される大きさは異なります
    - \* 最小値である 1x1 に設定しても実際に表示される大きさは 1x1 ではありません
  - 既定値は 640x360 です
- 出力背景色
  - ビューポートに表示する背景色を設定します
  - 透過度の設定が可能です
    - \* 透過度を 0 にすると黒背景になります
  - 既定値は白です ( R=255 / G=255 / B=255 / A=255)

#### 4.3.4 描画順ダイアログを開く

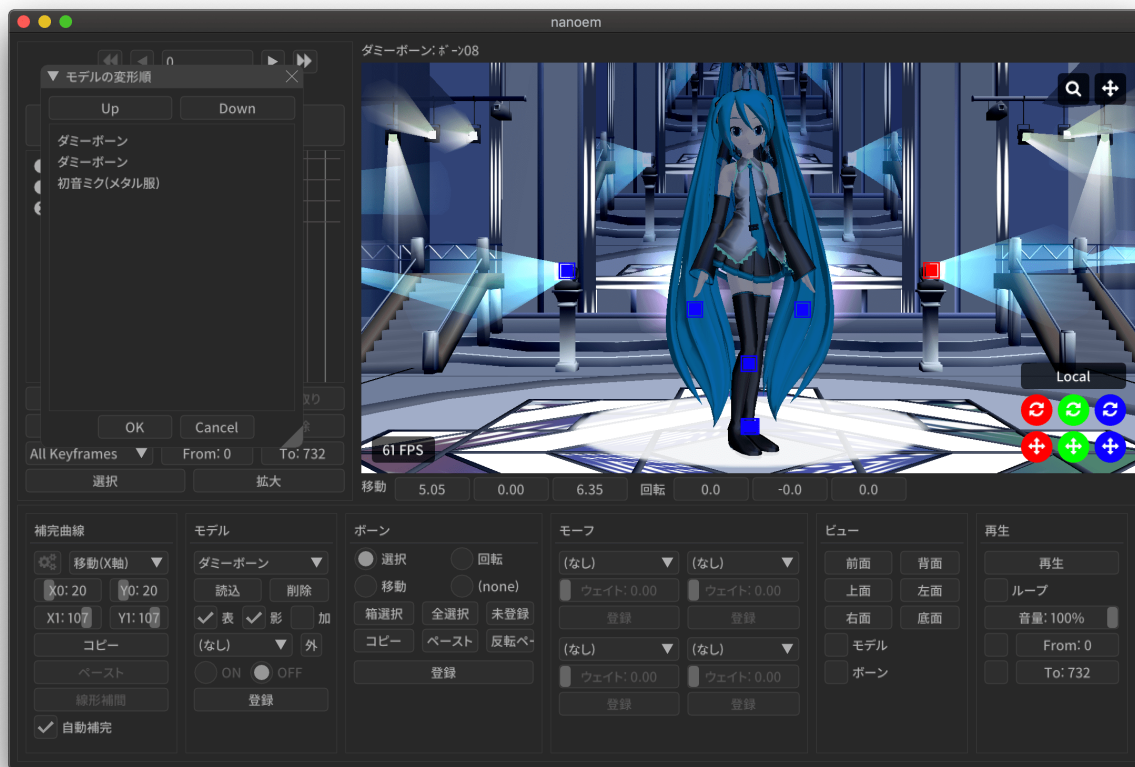


モデルの描画順を操作するダイアログを開きます。

- 描画順は上から順番に適用され描画されます
  - 画像の例では「ステージ」、「ダミーボーン」、「奥大赤右」の順番で描画されます
- Up ボタンを押すと選択対象を上へ移動（描画順が先になる）します
- Down ボタンを押すと選択対象を下へ移動（描画順が後になる）します

注釈：描画の仕様により、透過部分が含まれるモデルは描画順ダイアログを用いて後ろに描画するように調整する必要があります

### 4.3.5 変形順ダイアログを開く



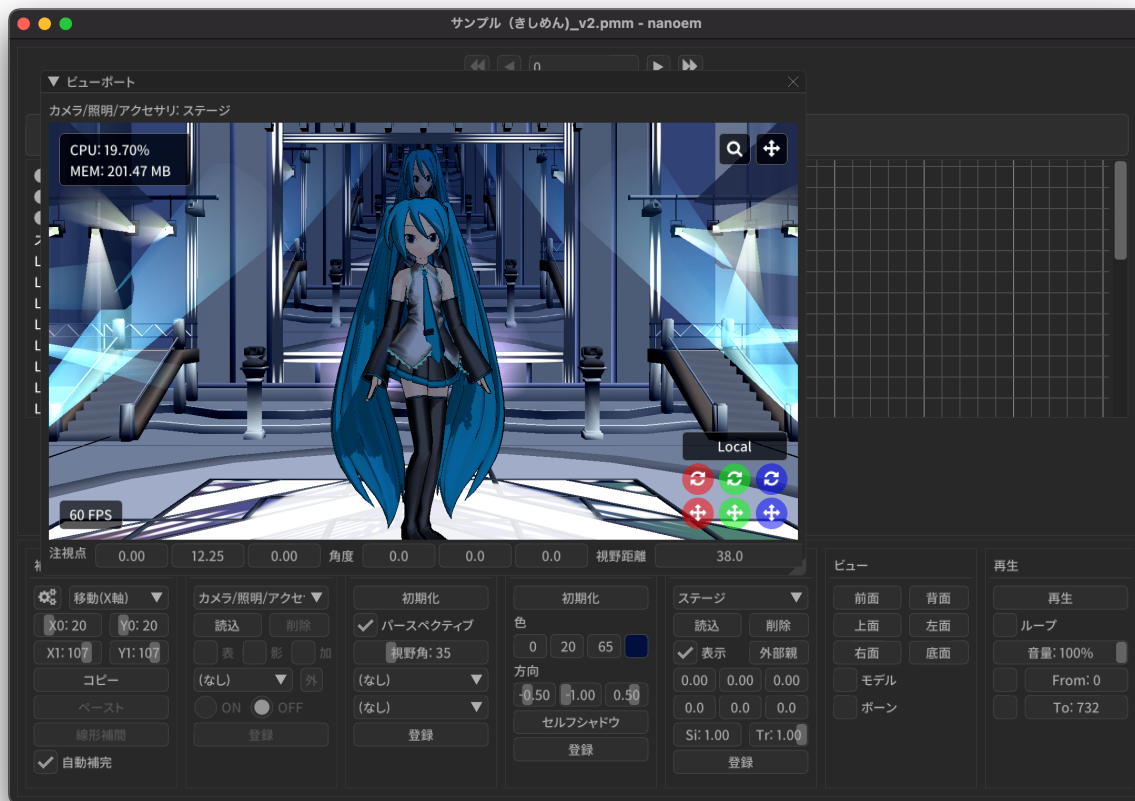
モデルの変形順を操作するダイアログを開きます。

- 変形順は上から順番に変形されます
  - 画像の例では「ダミーボーン」、「初音ミク（メタル服）」の順番で描画されます
- Up ボタンを押すと選択対象を上へ移動（変形順が先になる）します
- Down ボタンを押すと選択対象を下へ移動（変形順が後になる）します

ちなみに：この機能は外部親を設定する際に外部親先が先に変形するように順序を変えるためにあります。設定し忘れると外部親が正しくが行われなくなります。

### 4.3.6 ビューポートウィンドウを分離

ビューポート描画部分を独立したウィンドウとして分離します。



分離したビューポートウィンドウはウィンドウ右上の閉じるボタンで分離前の状態に戻すことができます。

#### 4.3.7 地面軸を有効にする

地面に設定された軸を表示するかを設定します。初期値は有効です。

#### 4.3.8 地面影を有効にする

---

**重要:** 地面影はセルフシャドウとは別です

---

地面に出来る影の表示を有効にするかを設定します。初期値は有効です。

#### 4.3.9 エフェクトを有効にする

MME 形式のエフェクトを有効にするかを設定します。初期値は無効です。



無効の場合はエフェクトを伴うアクセサリまたはモデル読み込み時に有効にするかどうかのダイアログが表示されます。

警告: バージョンによってエフェクトが使えたり使えなかったりする上に適用後の見た目が変わることがあります。適用後の描画が正しいかどうかはエフェクト配布元の適用後参考画像を確認したほうがよいです

#### 4.3.10 アンチエイリアス設定

ご用心: 有効にするとハードウェアによっては目に見えて重くなる場合があります。その場合は無効にしてください。

また、エフェクトによってはアンチエイリアス機能と干渉する場合があります。通常はエフェクトの取扱説明書にアンチエイリアスを無効化する必要があるかどうか書かれていますので、取扱説明書に明示的に無効にする必要がある場合において無効にしてください。

アンチエイリアスを設定します。初期値は無効です。

有効かつ数値が大きければ大きいほどざらつきが減ってきれいになりますが、その分処理負荷が上がります。また、最高設定にしても利用不可能な場合は利用可能な最も近い設定が自動的に選択されます。(例えば Apple M1 の場合は「最高設定」にしても「やや高」が選ばれる)

- 最高 (MSAAx16)
- 高 (MSAAx8)
- やや高 (MSAAx4)
- 普通 (MSAAx2)
- 無効

33.0 以降では無効以外に設定した場合はビューポートに表示されるボーン表示などのアンチエイリアスが有効になります。無効になっている場合でも「[高解像度ビューポートを有効にする](#)」を無効にしない限りは設定が維持されます。

#### 4.3.11 物理演算

物理演算の有効設定を切り替えます。有効設定は以下の 4 種類あります。

- 常時有効
  - 編集及び再生での物理演算を有効にします



- \* フレーム位置を前に移動した場合は演算をリセットします
- \* ボーン移動時でも物理演算が行われます
- 再生時のみ
  - 再生のみ物理演算を有効にします
  - 編集の場合は物理演算が無効になります
- トレースモード
  - 編集の場合 1 フレーム移動する毎に 1 フレーム分の物理演算を行います
    - \* フレーム位置を前に移動した場合は演算をリセットします
    - \* ボーン移動時は「常時有効」と違い、物理演算が行われません
  - 再生時は再生前に一回物理演算をリセットした後に有効にします
- 無効
  - 編集及び再生での物理演算を無効にします

## 演算設定

物理演算の演算設定を行います。

- 加速度
  - 重力の加速度を設定します
  - 加速度に方向を掛けた結果が重力設定として扱われます
- 方向
  - 重力方向を設定します
  - 浮遊する世界を表現したい場合は Y 軸をマイナスではなくプラスにすると重力が上向きになります
- 時間ステップ係数
  - 1 フレームあたりの物理演算のタイムステップ係数を設定します
  - 通常は設定する必要はありませんが、再生時フレームレートと動画書き出しの FPS が異なる場合はこの値を調整する必要があります
  - 例えば再生時フレームレートが 60FPS で動画書き出し想定が 120FPS の場合は引き伸ばしに必要な時間が 2 倍になるため 0.5 に設定します
    - \* このときタイムラインにある「拡大」を使ってモーションの長さを 2 倍に引き伸ばす必要があります

- \* 動画書き出し自体は 60FPS になりますが、引き伸ばすことで擬似的に 120FPS を実現できます。これを 5 分の 1 に縮小することで 24FPS にできます

### 全てのモデルのモーションを焼き込む [âRl](#)

---

重要: モーション焼き込みを実行後に焼き込み前の状態に戻せなくなる事故を防ぐためプロジェクトの上書き保存でも必ず保存ダイアログが表示されます (27.2.0 以降より適用)

---

プロジェクト内にいる全てのモデルのモーションに対して物理演算の結果を焼き込みをします。

- 「トレースモード」を有効にしたときと同じ処理結果になります
- 全ての剛体に紐づくボーンのキーフレームが 1 フレームずつ登録されるため、書き出されるモーションが肥大化します
- 焼き込みを行ったモーションの書き出しは vmd/nmd 両方可能です
  - ただし vmd の仕様上物理演算を切ることができないため読み込む側で物理演算を無効にする必要があります
  - またキーフレームの登録仕様上キーフレーム数上限に引っかかって MMD で読み込めない場合があります
    - \* nanoem ではキーフレーム数制限を設けていないためメモリが許す限り読み込むことができます
- 巻き戻す際は一回カメラモードに戻ってから実行する必要があります

### 全てのモデルのモーションを焼き込む (IK 含む) [âRl](#)

---

重要: モーション焼き込みを実行後に焼き込み前の状態に戻せなくなる事故を防ぐためプロジェクトの上書き保存でも必ず保存ダイアログが表示されます (27.2.0 以降より適用)

---

全てのモデルのモーションを焼き込む [âRl](#) と同じですが、こちらは IK を含めて焼き込みを行います。

- IK ボーンは全て無効になります
- IK 影響下のボーンのキーフレームがフレーム単位で登録されます
  - 結果として通常のモーション焼き込みと比較してさらにファイルサイズが大きくなります
- IK 影響下のボーンが付与親ボーンによって動かされる場合は IK 影響下のボーンではなく付与親ボーンに対して焼き込み処理を行います

- 巻き戻す際は一回カメラモードに戻ってから実行する必要があります

#### デバッグ描画

物理演算でのデバッグ描画を有効にします。Bullet Physics にあるデバッグ描画機能をそのまま用いています。

- ワイヤー
- AABB
- 接続点
- コンストレイント
- 制約角度

#### 4.3.12 再生時フレームレート

再生中におけるフレームレートを設定します。60FPS にするとモーションに補間が入り、滑らかに動くように調整されます。この設定は動画出力でも適用されます。

- 無制限
- 60FPS
- 30FPS

---

注釈: 「無制限」は垂直同期が無効化されること以外 60FPS と同等です

---

#### 4.3.13 音源を消去

現在読み込まれている音源を削除します。

また、プロジェクトの再生時間が再計算され最も長いモーションを再生時間として設定されます。

#### 4.3.14 背景動画を消去

現在読み込まれている背景動画を削除します。

「音源を消去」と異なりプロジェクトの再生時間の再設定は行われません。

#### 4.3.15 高解像度ビューポートを有効にする

---

注釈: macOS では高解像度設定でのみ有効です

---

高解像度設定時の高解像度ビューポートの切り替えをします。

高解像度液晶はビューポートに対する表示負荷が大きいため、表示負荷を下げるときに利用 (= 無効) します。解像度の倍率によりますが、例えば 2 倍の場合は無効にすることによりビューポートの描画が荒くなるかわりに表示負荷を 1/4 に減らすことが可能です。

33.0 以降ではビューポートの高解像度化に加えてビューポートに表示されるボーン表示などのアンチエイリアスが有効になります。高解像度ビューポートが無効になっている場合でも「[アンチエイリアス設定](#)」を無効にしない限りは設定が維持されます。

#### 4.3.16 FPS カウンターの表示を有効にする

ビューポート画面左下に表示される FPS カウンターの表示を切り替えをします。初期値は有効です。

#### 4.3.17 パフォーマンスモニターの表示を有効にする

---

注釈: 複数の CPU が利用可能な場合 CPU 利用率が場合によっては 100% を超えるときがありますが、仕様です

---

ビューポート画面左上に表示されるパフォーマンスモニターの表示を切り替えをします。初期値は有効です。

パフォーマンスに表示される内容は以下です。

- 現時点での nanoem の CPU 使用率
  - Windows の場合は nanoem ではなくマシン自体の CPU 使用率になります
- 現時点での nanoem のメモリ消費量

### 4.4 カメラ

#### 4.4.1 プリセットから選択

プリセットからパラメータを設定します。いずれも角度のみを変更し、それ以外のパラメータは維持されます

- 上面

- 左
- 右
- 底面
- 前面
- 背面

#### 4.4.2 キーフレームを登録

現在のフレーム位置に対してカメラのキーフレームを登録します。

#### 4.4.3 キーフレームを削除

選択されたカメラのキーフレームを全て削除します。

#### 4.4.4 初期化

カメラのパラメータを「新規プロジェクト」時のパラメータに初期化します。

項目	値
位置	(X=0/Y=10/Z=0)
角度	(X=0/Y=0/Z=0)
視野角	30
視野距離	45

### 4.5 光源

#### 4.5.1 セルフシャドウ

モード選択

セルフシャドウの有効設定を切り替えます。以下の3種類あります。

- 無効
  - セルフシャドウを無効にします
- モード1 ( バランス型 )

- セルフシャドウを有効にした上で全体的にセルフシャドウを割り当てるようにします
- 遠景にもセルフシャドウをかけたい場合に有効です
- モード 2 (近接重視型)
  - セルフシャドウを有効にした上でカメラに近い対象オブジェクトにより多く割り当てるようにします
  - カメラ制御によりカメラアップを使う場合に有効です

### キーフレームを登録 [âRl](#)

現在のフレーム位置に対してセルフシャドウのキーフレームを登録します。

### キーフレームを削除 [âRl](#)

選択されたセルフシャドウのキーフレームを全て削除します。

### 初期化

セルフシャドウのパラメータを「新規プロジェクト」時のパラメータに初期化します。

項目	値
モード	モード 1
距離	8875

## 4.5.2 キーフレームを登録 [âRl](#)

現在のフレーム位置に対してカメラのキーフレームを登録します。

## 4.5.3 キーフレームを削除 [âRl](#)

選択された光源のキーフレームを全て削除します。

## 4.5.4 初期化

光源のパラメータを「新規プロジェクト」時のパラメータに初期化します。

項目	値
色	(R=153/G=153/B=153)
方向	(X=-0.5/Y=-1.0/Z=0.5)

## 4.6 モデル

### 4.6.1 モデルを選択

プロジェクト内に読み込まれているモデルから操作するモデルを選択します。

### 4.6.2 ボーンを選択

選択されているモデルから操作対象とするボーンを選択します。モデルが選択されていない場合は選択出来ません。

「選択モード」でボーンを選択するのと同等の処理です。

### 4.6.3 モーフを選択

選択されているモデルから操作対象とするモーフを選択します。モデルが選択されていない場合は選択出来ません。

モーフは以下のカテゴリに分別されそれぞれのカテゴリに属するモーフから選択します。

- 目
- まゆ毛
- リップ
- その他

### 4.6.4 エッジ設定

選択されているモデルに対するエッジを設定するダイアログを開きます。以下の2項目を設定することが可能です。

- エッジ色
- エッジ幅

---

注釈: キーフレームを登録しないと設定が反映されないため、確定する場合は必ず「キーフレーム登録」をしてください (内部的にはモデルのキーフレーム登録として実行されます)。

---

ご用心: VMD の仕様上エッジ設定を保存することが出来ません。そのため VMD 書き出し時に情報が失われます。

#### 4.6.5 モデルの身長測定

---

注釈: 言語設定が英語の場合のみフィート及びインチ単位の測定結果が追加表示されます (モデル編集では言語設定に関わらずセンチメートル単位固定です)。

---

選択されているモデルの身長の近似値をセンチメートル単位で計測します。

モデルの身長測定の計算方法はモデル編集の「9CD37758-C019-4447-B85A-D2306A90C44E」にあるものと同じですが、デフォルト値固定で実行されます。モデル編集機能を有効にしなくても利用可能です。

#### 4.6.6 モデルデータの検証を実行

---

注釈: モデル編集有効時に実行するとモデル編集で発生する問題対処のためにより多くの項目が検証されます。

---

選択されているモデルに対する検証処理を実行します。

主にテクスチャファイルが見つからないことを起因するモデル表示トラブルを利用者が把握できるようにする目的で利用します。

#### 4.6.7 全てのトラックを展開

タイムラインにあるトラックを全て展開します。

#### 4.6.8 全てのトラックを折りたたみ

タイムラインにあるトラックを全て折りたたみます。



#### 4.6.9 加算ブレンドを有効にする

モデルの加算ブレンドを有効にします。モデル全体の明るさを上げる効果があります。

#### 4.6.10 シャドウマップを有効にする

シャドウマップを有効にしてセルフシャドウを表示させるようにします。

#### 4.6.11 表示を有効にする

---

注釈: 33.0 以降では非表示モデルに対する物理演算も無効になります

---

モデルの表示を有効にします。無効にするとモデルが表示されなくなりますが、その分描画がはやくなります。

#### 4.6.12 キーフレームを登録 [â†’](#)

モデルのキーフレームを登録します。これはボーンやモーフのキーフレームと別枠です。

モデルのキーフレームで登録される情報は以下です。

- モデルの表示状態
- モデルの IK 状態

#### 4.6.13 選択されたキーフレームを削除 [â†’](#)

選択されたモデルのキーフレームを削除します。これはボーンやモーフのキーフレームと別枠です。

#### 4.6.14 モデル設定

危険: 「モデル設定」は説明を見て理解出来なければ操作するべきではない

全てのボーンを表示する

非表示のボーンを含めて全て表示します。通常「剛体に紐付いたボーン」は直接操作することはないため非表示ボーンとして表示されませんが、チェックを入れる则表示されるようになります。場合によってモデルの表示が崩れることがあります。

#### 剛体を表示する

物理演算の剛体を表示します。場合によってモデルの表示が崩れることがあります。

#### 頂点の面を表示する

モデルの頂点の面を全て表示します。場合によってモデルの表示が崩れることがあります。

#### 頂点の点を表示する

モデルの頂点の点を全て表示します。場合によってモデルの表示が崩れることがあります。

### 4.6.15 現在選択中のモデルを削除

ご用心: この処理は巻き戻すことができません

現在選択されているモデルをプロジェクトから削除します。また、対象モデルのモーションも一緒に削除されます。

モデルをプロジェクトから外す形の処理のため、読み込み元のモデルのファイルは保持されます。

## 4.7 アクセサリ

### 4.7.1 キーフレームを登録

現在のフレーム位置に対してアクセサリのキーフレームを登録します。

### 4.7.2 選択されたキーフレームを削除

選択されたアクセサリのキーフレームを全て削除します。

### 4.7.3 加算ブレンドを有効にする

アクセサリの加算ブレンドを有効にします。初期値は無効です。

### 4.7.4 地面影を有効にする

**重要:** アクセサリは仕様として地面影のみでセルフシャドウを描画することが出来ません。セルフシャドウを使いたい場合はモデルに変換する必要があります。

選択しているアクセサリの地面影を有効にします。初期値は無効です。

#### 4.7.5 表示を有効にする

選択しているアクセサリの表示を有効にします。初期値は有効です。

#### 4.7.6 リセット

アクセサリを読み込み直後のパラメータに初期化します。

項目	値
位置	(X=0/Y=0/Z=0)
回転	(X=0/Y=0/Z=0)
Si (拡大率)	1
Tr (不透明度)	1

#### 4.7.7 現在選択中のアクセサリをモデルに変換

注釈: モデル編集用途の機能のためデフォルトでは表示されません。利用する場合は「[アプリケーションの設定](#)」から「[モデル編集機能を有効にする](#)」に ☒ を入れる必要があります。

変換したモデルは以下の仕様に従って作成されます。見た目は必ずしも一致するとは限らないため材質調整が必要な場合があります。

- 面は全て三角形に変換
- 材質数は元のアクセサリと一致
- 「全ての親」ボーンのみ存在、表示枠も同様
- 全頂点が「全ての親」に対して BDEF1 でマッピング
- スフィアマップの拡張子 (sph/spa) が指定されている場合はスフィアマップとして設定
- モーフ、剛体、ジョイント、ソフトボディはそれぞれ空要素

ご用心: この処理は巻き戻すことができません

現在選択中のアクセサリをモデルに変換します。この処理は以下の流れで行われます。

- 読み込み元アクセサリファイルからモデルに変換
- 変換したモデルをアクセサリのファイル名から拡張子 `pmx` に変更して保存
- 先のファイルを読み込んで新規モデルとして追加
- 元のアクセサリのモーションのうち「位置」と「回転」のみを「全ての親」ボーンキーフレームに登録する形でコピー
- 変換したアクセサリをプロジェクトから削除

元のアクセサリはプロジェクトから削除されますが、読み込み元アクセサリのファイルは維持されます。

#### 4.7.8 現在選択中のアクセサリを削除

ご用心: この処理は巻き戻すことができません

現在選択されているアクセサリをプロジェクトから削除します。また、対象アクセサリのモーションも一緒に削除されます。

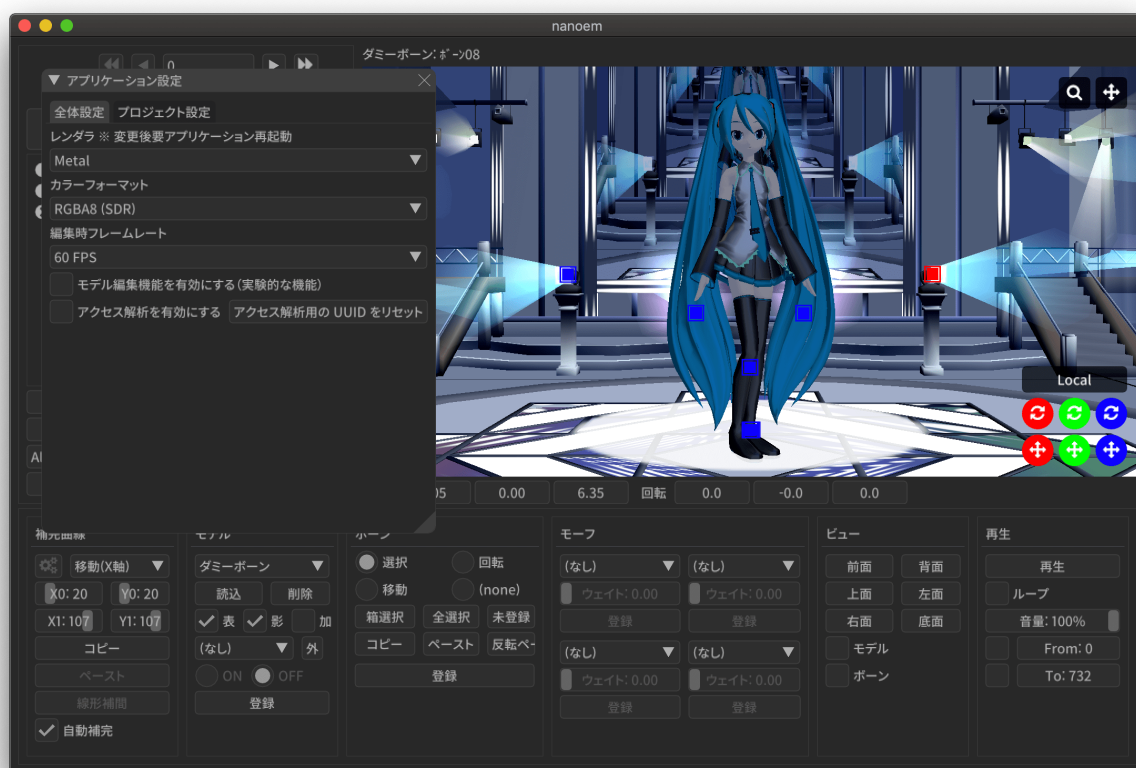
アクセサリをプロジェクトから外す形の処理のため、読み込み元のアクセサリのファイルは保持されます。

## 第 5 章

# アプリケーションの設定

- macOS 版の場合はメニューから「nanoem」>「設定」から開く
- Windows 版の場合はメニューから「編集」>「設定」から開く

### 5.1 全体設定



### 5.1.1 レンダラ

重要: macOS Mojavi 以降 OpenGL が非推奨に指定されたため、Metal が利用可能な場合は Metal を自動的に選択します。

nanoem で使用するレンダラを変更します。OS によって利用可能なレンダラが異なります。

レンダラ	MacOS で利用可能 ?	Windows で利用可能 ?
OpenGL	はい(Y)	はい(Y)
Metal	はい(Y)	いいえ(N)
DirectX11	いいえ(N)	はい(Y)

macOS 版の場合は利用可能なら Metal を、Windows 版は DirectX11 をデフォルトとして利用します。

### 5.1.2 カラーフォーマット

ビューポート表示で使うフォーマットを表示します。デフォルトは RGBA8 です。

- RGBA8
- RGB10A2
- RGBA16F

RGB10A2 及び RGBA16F は HDR のため、正しく表示するには HDR に対応する外部液晶が必要です。また、RGBA16F を選択すると RGBA8 よりも描画速度が遅くなります。

macOS 版かつ RGBA16F を選択したときのみ利用する色空間が Display P3 に変更されます。

### 5.1.3 編集時フレームレート

注釈: プロジェクト再生時はメニューの「プロジェクト」の「フレームレート」が適用されます。

プロジェクトを再生している時以外の FPS を選択します。

- 自動
  - 状況に応じて 30FPS または 60FPS を切り替えつつ処理します
  - \* 電源接続時はパフォーマンスを高めるため 60FPS で処理されます

\* バッテリー駆動の場合は消費を抑えるため 30FPS で処理されます

- 30FPS
- 60FPS
- 無制限
  - 垂直同期が無効になるので 60FPS を超えて最大出力で処理されます
  - 非常に重くなるので、どうしてもという時以外は使用しないこと

#### 5.1.4 高解像度ビューポート

注釈: この設定は高解像度液晶を採用しながら相対的に描画性能が厳しい外付け GPU がない Intel 版の MBA/MBP をターゲットとしています。ビューポートだけ描画が荒くなりますが、描画負荷を抑えることが可能になります (特にポストエフェクト利用で効果が大きく現れます)。

高解像度ビューポートの設定方法を選択します。あくまで起動時の設定なので、「高解像度ビューポートを有効にする」で切り替えることが可能です。

- 起動時に自動設定
  - 起動時に外付け GPU が選択されている場合は有効に、それ以外の場合は無効にします
  - 初期設定は「起動時に自動設定」が適用されます
- 常に有効
- 常に無効
  - 29.0 未満は「常に有効」相当の処理が適用されていました

#### 5.1.5 モデル編集機能を有効にする

ご用心: 当該機能は実験的な機能のため、今後仕様が変更される可能性があります。

注釈: 「モデルプラグイン」の利用において「モデル編集機能を有効にする」は不要です。

以下の機能を有効にして利用可能にします。

- model
- メニューにモデル編集機能関連の機能を追加
  - モデル (PMX 形式) の書き出しも含まれる

初期値は無効です。

### 5.1.6 モデル表示の負荷分散を有効にする

---

注釈: 設定の反映はプロジェクトの新規作成または一回保存して読み込みする必要があります。

---

モデル表示の際に CPU で演算される高負荷な処理であるスキニングを GPU に分散させます。有効にすると CPU 負荷が軽減されます。ただし GPU によって演算結果が意図した結果にならずモデル表示が崩れることがあるため、その場合は無効にする必要があります。

負荷分散が有効になるのはあくまでモデルのスキニングのみなので、アクセサリやモデルの物理演算には適用されません。また、以下の場合も「[モデル表示の負荷分散を有効にする](#)」の状態に関係なくすべて CPU で演算します。

- ソフトボディがひとつ以上含まれたモデル利用時
  - 剛体がボーン単位であるのに対してソフトボディが頂点単位処理で競合するため高速化の恩恵を受けづらいためです
  - ソフトボディのみで構成されたモデルを分けて「[外 \(外部親設定\)](#)」を利用することである程度対処できます
- 「model」モード時
  - モード切替前に有効だった場合はモード終了時に無効から有効に戻します

初期値は無効です。

### 5.1.7 クラッシュレポートを有効にする

---

注釈: 設定の反映をするにはアプリケーションの再起動が必要です。

---

クラッシュレポートを有効にします。クラッシュレポートは [Sentry](#) を利用しています。

初期値は有効です。



### 5.1.8 アクセス解析を有効にする

---

重要: アクセス解析は 32.0 で廃止されたため現在は機能しません。

---

---

注釈: 設定の反映をするにはアプリケーションの再起動が必要です。

---

アクセス解析を有効にします。アクセス解析は [Google Analytics](#) を利用しています。

初期値は有効です。

### 5.1.9 アクセス解析用の UUID をリセット

---

注釈: 処理の関係上設定の反映をするにはアプリケーションの再起動が必要です。

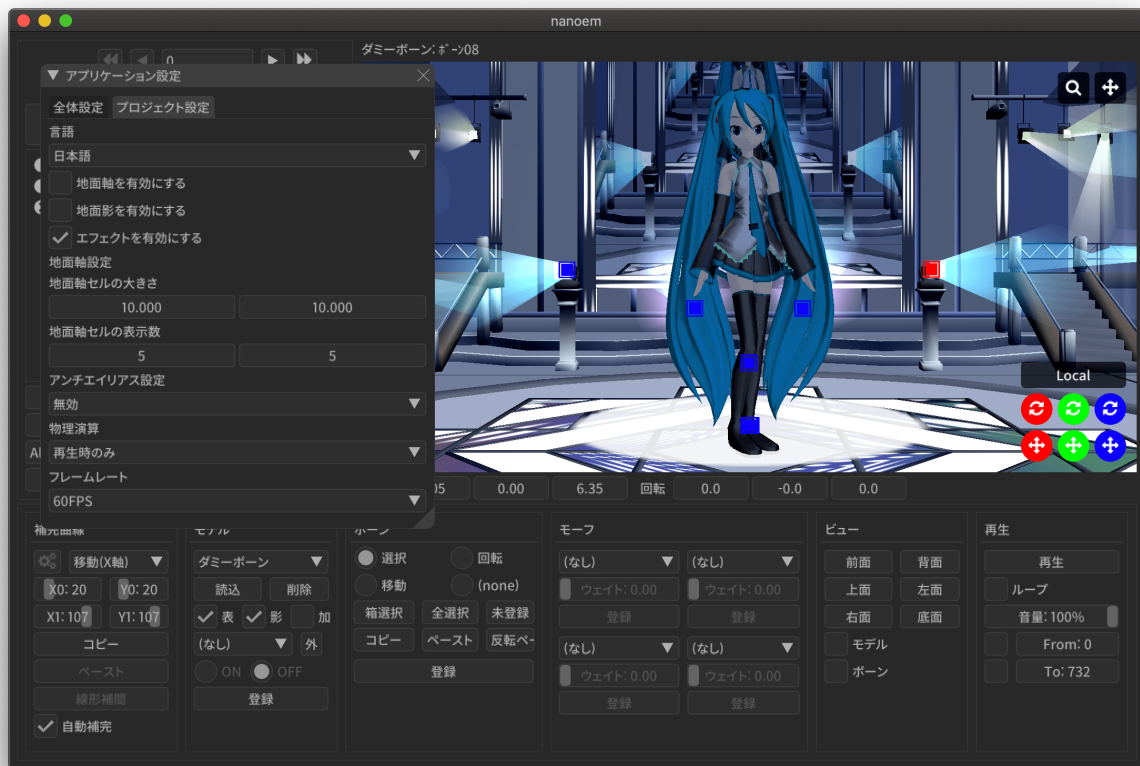
---

アクセス解析用の UUID をリセットします。サポートから開発者より要請を受けた場合にのみ使用してください。

### 5.1.10 システム情報

nanoem が動作している CPU 名、プラットフォーム名、ビルド情報を見ることができます。

## 5.2 プロジェクト



### 5.2.1 言語

ご用心: モデル編集での言語変更はできません。変更する場合は一回モデル編集を終了する必要があります。

注釈: メニューを含め即座で変更が反映されるため、アプリケーションの再起動は不要です。

UI で表示する言語を変更します。以下から選択可能です。

- 日本語
- 英語

デフォルトは OS の言語設定に依存します。OS の言語設定が日本語の場合は「日本語」が、それ以外の場合は「英

語」が選ばれます。

### 5.2.2 地面軸を有効にする

「地面軸を有効にする」と同じです。

### 5.2.3 地面影を有効にする

「地面影を有効にする」と同じです。

### 5.2.4 エフェクトを有効にする

「エフェクトを有効にする」と同じです。

### 5.2.5 地面軸設定

地面軸セルの大きさ

地面軸のセル（格子）の大きさを変更します。

地面軸セルの表示数

地面軸のセル（格子）の表示数を変更します。多ければ多いほど描画が重くなります。

### 5.2.6 アンチエイリアス設定

「アンチエイリアス設定」と同じです。

### 5.2.7 物理演算

「物理演算」と同じです。

### 5.2.8 フレームレート

「再生時フレームレート」と同じです。

## 5.2.9 保存時のファイルパス設定

---

注釈: ファイルパスとは以下の種類の読み込まれたファイルのパスを指します。これらはプロジェクトファイルに中身を含めないためです。

- モデル
  - アクセサリ
  - 材質割当したエフェクト
- 

nmn 形式で保存する際のプロジェクト内に保存するファイルパスの設定の変更を行います。

初期値は相対パスですが、29.2.0 以前のバージョンでは絶対パスで保存されていた関係で当該バージョンで保存されたプロジェクトを読み込むと絶対パスが初期値になります。

### 絶対パス

プロジェクトファイルの保存位置に依存しないファイルまでの完全なパスで保存します。

メリットはプロジェクトファイル自体を移動させた場合でも読み込むことができます。

デメリットとして例えばオンラインストレージなどで保存して別のマシンに同期後動かすと全く同じパスでないと読み込めなくなります。また、読み込んだモデルやアクセサリを移動させると同じく読み込めなくなります。さらに保存場所によりますが完全なパスを構築する関係でユーザ名のフォルダ名が含まれることが多く個人情報が見えてしまうためファイルの受け渡しに向いていません。

### 相対パス

ご用心: 相対パスで設定しても相対パスが解決出来なかった場合は絶対パスで保存されます。これは Windows においてプロジェクトの保存先と異なるドライブで読み込まれた場合に該当します。

プロジェクトファイルの保存位置からファイルまでの相対的なパスで保存します。

メリットは例えばオンラインストレージなどで保存して同期した後に別のマシンで動かすことが可能になります。

デメリットとしてプロジェクトファイル自体を移動させた場合は読み込んだモデルやアクセサリも全て同時に移動させる必要がある点です。また、読み込んだモデルやアクセサリを移動させると同じく読み込めなくなります。

## 第 6 章

# エフェクトの設定

エフェクト設定ウィンドウを開く から開くことのできる個々のエフェクト設定画面です。

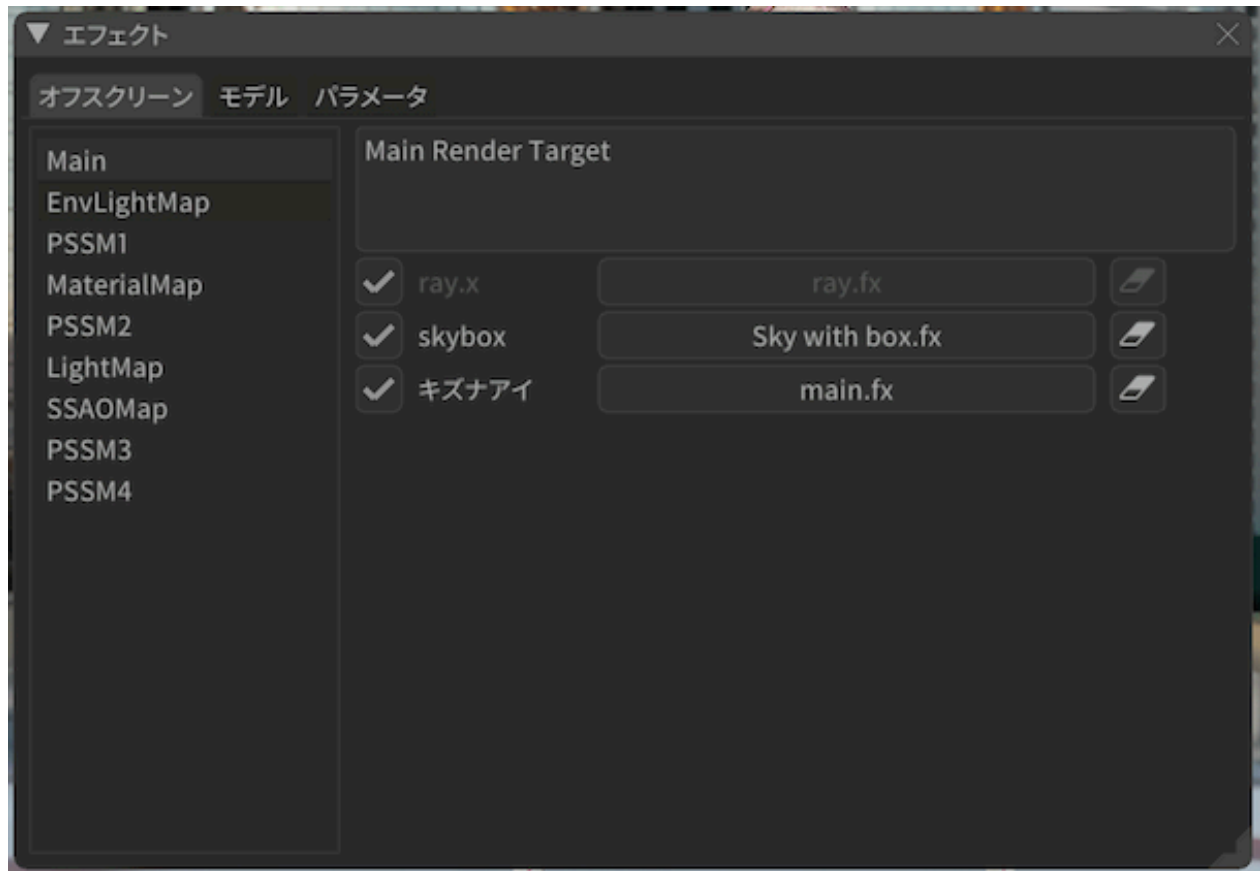
主に個々のエフェクトの割り当てを変える時に利用します。

---

注釈: スクリーンショットで利用しているものについて

- モデルは [キズナアイ](#) を利用しています
  - エフェクトは [Ray-MMD \(1.5.2\)](#) を利用しています
-

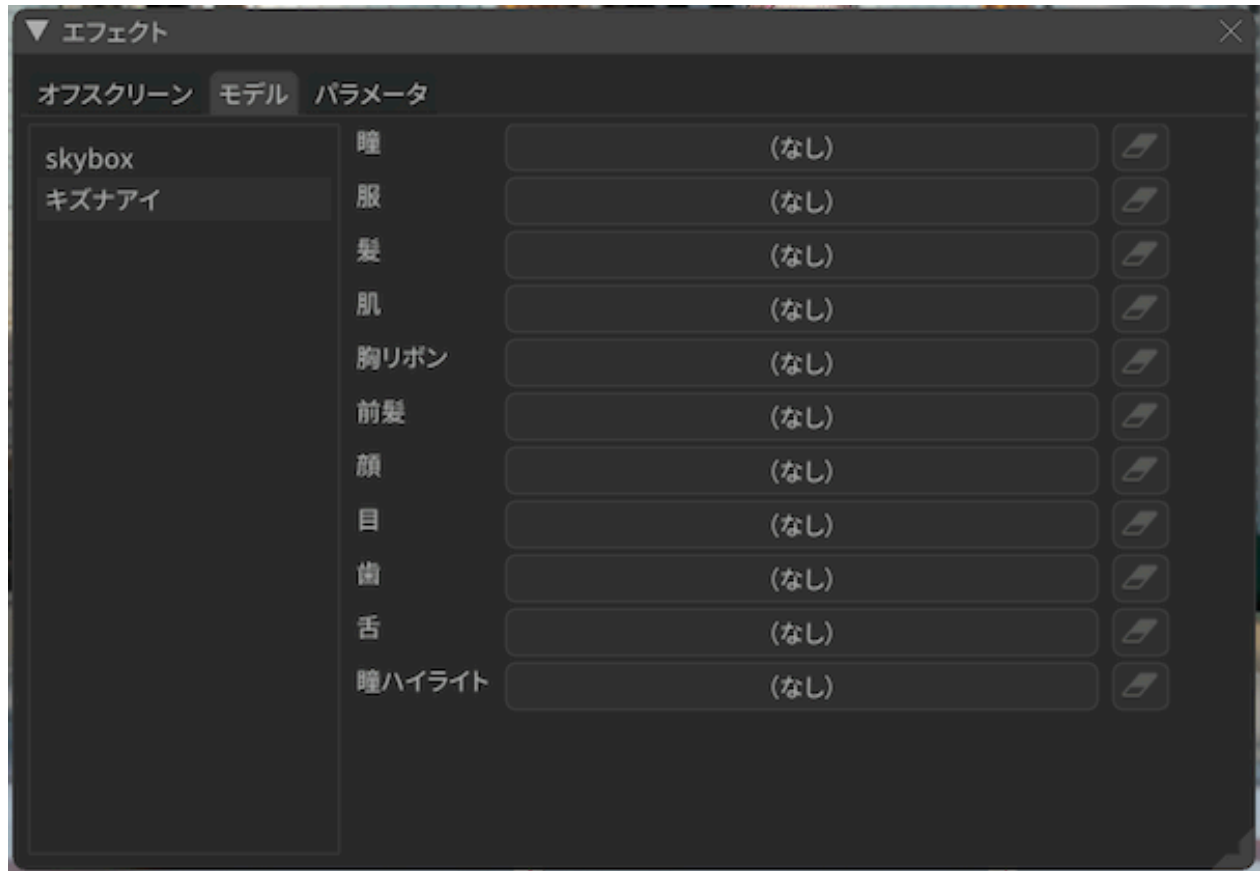
## 6.1 オフスクリーン



オフスクリーンのエフェクト割り当て設定を変えることができます。

- 画面左側にはプロジェクト内で管理しているオフスクリーンエフェクトの一覧が出ます。
- 画面右側は選択されたオフスクリーンの詳細が出ます。
  - 上部にオフスクリーンの説明が表示されます。
  - 下部にはオフスクリーンに適用されているモデルまたはアクセサリの一覧が出ます。
    - \* ファイル名が表示されたボタンをクリックするとファイルダイアログが表示され、割り当てるエフェクトを変えることができます
    - \* 消しゴムボタンをクリックすると割り当てを解除することができます
    - \* グレーアウトされている部分は操作不可能です（例えばオフスクリーン自身のモデルまたはアクセサリである場合）

## 6.2 モデル



モデルの材質単位に割り当てるエフェクトの設定を変えることができます。

**重要:** 30.3.0 から MME で使われているモデルのエフェクト割当情報設定ファイルである emd 形式のファイルの読み書きが実装されました。

- 画面左側にはプロジェクト内で管理しているモデルまたはアクセサリの一覧が出ます。
  - 画面左下の「読み」は emd 形式のデータを読み込みます
  - 画面左下の「保存」は emd 形式のデータを保存します
- 画面右側はモデルの材質に割り当てるエフェクト一覧が出ます。
  - ボタンの部分をクリックするとファイルダイアログが表示され、割り当てるエフェクトを変えることができます
  - 消しゴムボタンをクリックすると割り当てを解除することができます

## 6.3 パラメータ

パス、レンダーターゲットの一覧を表示します。「エフェクトをリロードする」ボタンを押すと選択しているモデルまたはアクセサリのエフェクトをリロードすることができます。

ヒント: エフェクトのソースコードを書き換えると変更を検知して自動的にリロードする仕組みを持ってるのでそちらの方が簡単です。

ただし `include` されたエフェクトの自動検知は対応していませんので `include` している側のエフェクトを書き換える必要があります (主に拡張子が `conf` や `fxsub` のファイルが `include` される側のエフェクトです)。



「パラメータのデバッグを有効にする」を有効にするとエフェクトに割り当てられるパラメータの中身を表示します。





レンダーターゲットの場合は上部に説明、下部にレンダーターゲットの中身を画像として表示します。



## 第 7 章

# プラグイン

---

重要: プラグインを使用するには 29.2.0 以上が必要です。またプラグインの提供及び導入方法が今後変更される可能性があります。

---

### 7.1 導入方法 (macOS)

- nanoem アプリを Finder から「パッケージの内容を表示」で開く
- Contents > Plugins に移動し Plugins フォルダにプラグインをコピー
  - 例えば「準標準ボーンプラグイン」の場合は `plugin_ssb.dylib` を Plugins フォルダにコピーする
- nanoem を再起動
- メニューから「編集」>「モデルプラグイン」に導入したいプラグインの名前が出ることを確認する
  - 例えば「準標準ボーンプラグイン」の場合「準標準ボーンプラグイン」が名前に出る

### 7.2 準標準ボーンプラグイン

「準標準ボーンプラグイン」(macOS 版) をダウンロード

---

重要: 「準標準ボーンプラグイン」は配布モーションによって必須指定するほどの重要性の高いプラグインのため特例として移植して配布していますが、原則として PMDEditor/PMXEditor 系のプラグインの移植はいたしません。予めご了承ください。

Windows 版の配布予定はありません (そもそも PMDEditor/PMXEditor が使える環境でありわざわざ nanoem 側のプラグインを使う理由がないため)。

Semi-Standard Bone Plugin として英語表記に対応しています (移植元のオリジナルの方は英語表記に対応していません)。

---

警告: 「標準ボーンプラグイン」の利用はモデルの改造行為にあたるため、プラグイン利用前にモデルの利用規約を確認してください。ただしモデル配布の段階で標準ボーンプラグインで作成されるボーンが最初から入っていることがありその場合は実行不要です。

---

注釈: 「標準ボーンプラグイン」含むモデルプラグインを実行すると元のモデルに対して上書き保存します。そのため、実行前のモデルデータのバックアップが作成されて対象モデルと同じフォルダに [モデルのファイル名]\_backup\_at\_[年月日]\_[時刻].pmx として書き出されます (例えばモデルのファイル名が「初音ミク」で実行時刻が「2020 年 7 月 7 日 12 時 39 分 51 秒」の場合作られるファイル名が 初音ミク\_backup\_at\_20200707\_123951.pmx になります)。

実行元が PMD の場合は 必ずバックアップを取ってください。内部の実装仕様上モデル読み込み時に PMX に変換するためです。

モデルをプラグイン実行前のデータに元に戻したい場合はバックアップを元のモデルファイルにコピーしてください。なお、バックアップ作成を無効にすることはできません。

---

「標準ボーンプラグイン」はそぼろさんが開発する PMDEditor/PMXEditor 用のプラグインである「標準ボーンプラグイン」とほぼ同じ機能を実現する [モデルプラグイン](#) です。

[そぼろさんによる標準ボーンプラグインの解説動画はこちら](#)

これは標準モデルである 初音ミク.pmd に存在しないものの以下の理由からに広く使われるボーンを自動的に作成するものです (作成されるボーンがすでにある場合は処理をスキップします)。

- モーションを作りやすくするため
- モーションの動きの自由度を上げるため
- 配布されてるモーションに直接手を加えることなく補正できるようにするため
  - 回転付与のボーンは主にこの用途です

配布されるモーションによっては標準ボーンプラグインで作られるボーンを必須とすることがあります (なくても実際には読み込めるもののモーション作成者が意図しない動きが発生する可能性がある)。

- \* が後ろについてるものは頂点のボーン再配置及びウェイト塗り (以降自動再配置) を実行します

- aŽāyŕ が先頭についてる場合「必要なボーン」が足りてないため実行するとエラーが発生することを示しています
  - 上記によるエラーが発生した場合はモデルが変更されずそのままになります
  - 「必要なボーン」は「日本語」のボーン名で判定します
  - ボーン名の判定において全角英数字と半角英数字を区別します

### 7.2.1 腕擦れボーン \*

「右腕擦」と「左腕擦」を作成します。それぞれ「右腕」と「右ひじ」、「左腕」と「左ひじ」の間に作成されます。

また、擦ボーンを親とする回転付与率が 0.25/0.5/0.75 の 3 つの回転付与ボーンが作成されます。

ちなみに「擦れ」は「ねじれ」と読みます。

必要なボーン

- 右腕
- 右ひじ
- 左腕
- 左ひじ

### 7.2.2 回転自動補正

「腕擦れボーン」作成時に腕にマッピングされている頂点を元に位置補正を行うかを指定します。

### 7.2.3 手擦れボーン \*

「右手擦」と「左手擦」を作成します。それぞれ「右ひじ」と「右手首」、「左ひじ」と「左手首」の間に作成されます。

また、擦ボーンを親とする回転付与率が 0.25/0.5/0.75 の 3 つの回転付与ボーンが作成されます。

ちなみに「擦れ」は「ねじれ」と読みます。

必要なボーン

- 右ひじ

- 右手首
- 左ひじ
- 左手首

### 7.2.4 上半身 2 ボーン \*

「上半身 2 ボーン」を作成します。これは「上半身」と「首」の間に作成されます。

上半身と首の間、すなわち胸部に位置するということもあって特に高い頻度で利用されるボーンであり、体の傾きを調整しやすくします。

必要なボーン

- 上半身
- 首

### 7.2.5 腰ボーン

「腰ボーン」を作成します。これは「下半身」と「足」の間に作成されます。

また「右足」と「腰」の間に「腰キャンセル右」が、「左足」と「腰」の間に「腰キャンセル左」が作成されます。名前が示す通り回転補正が -1.0 で行われる回転付与のボーンで、震えを制御するために使います。

必要なボーン

- 下半身
- 右足
- 左足

### 7.2.6 足 IK 親

---

注釈: 作成されるボーン「足 IK 親」の IK が全角ではなく半角英数字な点にご注意ください

---

「右足 IK 親」と「左足 IK 親」を作成します。これは「右足 I K」と「左足 I K」の親として作成されます。

## 必要なボーン

ご用心: 全角英数字の方の「 I K 」です。半角英数字の「 IK 」ではないのでご注意ください

- 右足 I K
- 左足 I K

## 7.2.7 足先 EX \*

「右足先 EX」と「左足先 EX」を作成します。加えて以下の回転付与のボーンが作成されます。

- 右足 D
- 右ひざ D
- 右足首 D
- 左足 D
- 左ひざ D
- 左足首 D

これらの回転付与のボーンは元のモーションに直接手を加えることなく追加の回転補正を行えるようにするために使われます。

## 必要なボーン

ご用心: 全角英数字の方の「 I K 」です。半角英数字の「 IK 」ではないのでご注意ください

- 右足
- 右ひざ
- 右足首
- 右つま先 I K
- 左足
- 左ひざ
- 左足首

- 左つま先 I K

### 7.2.8 足 D ボーンを操作可能に

ご用心: 「ボーン表示枠に自動登録」に `âIJĚ` を入れる必要があります

「足先 EX」で作られる末尾が D のボーンをタイムラインの表示枠に追加にするかどうかを指定します。

### 7.2.9 手持ちアクセサリ用ダミー

「右ダミー」と「左ダミー」を作成します。それぞれ「右手首」と「右中指 1 」、 「左手首」と「左中指 1 」の間に作成されます。

手に持たせる形のアクセサリを外部親経由で持たせるために使います。

必要なボーン

ご用心: 全角英数字の方の「1」です。半角英数字の「1」ではないのでご注意ください

- 右手首
- 右中指 1
- 左手首
- 左中指 1

### 7.2.10 肩キャンセルボーン

以下のボーンを作成します。P は右肩または左肩の親として、C は右肩または左肩と右腕または左腕の間に入ります。

- 右肩 P
- 右肩 C
- 左肩 P
- 左肩 C



P は親を、C はキャンセルを意味し、C の方は付与率が -1.0 の回転付与ボーンとして作成されます。

#### 必要なボーン

- 右肩
- 右腕
- 左肩
- 左腕

### 7.2.11 親指 0 ボーン \*

「右親指 0」と「左親指 0」ボーンを作成します。それぞれ「右手首」と「右親指 1」、「左手首」と「左親指 1」の間に作成されます。

#### 必要なボーン

ご用心: 全角英数字の方の「1」です。半角英数字の「1」ではないのでご注意ください

- 右手首
- 左手首
- 右親指 1
- 左親指 1

### 7.2.12 親指ローカル軸設定

親指 0 ボーン作成時に各親指ボーンのローカル軸を追加で設定します。

### 7.2.13 グループボーン

「グループ」ボーンを作成します。「センター」ボーンの直後に配置されます。

いわゆるボーンの多段化の一種であり、実質的に全ての動きを司るセンターボーンから上下の動きを分離するために使われます。

必要なボーン

- センター

### 7.2.14 全ての親

「全ての親」ボーンを作成します。最上位に設定されます。

名前が示す通り全てのボーンの親として設定され、個々のボーンを直接編集することなくモデル自体の位置を補正するために使われます。ただし、配布用に作る場合にかぎり先の理由で干渉を起こす可能性があることから利用を避けるべきです。

必要なボーンはありません。

### 7.2.15 操作中心

「操作中心」ボーンを作成します。最上位に設定（優先度的には「全ての親」よりも上）されます。

特殊な扱いを受けるボーンで [モデル 1.20 以降](#) でカメラの視点の中心に設定するために使われます（モデルを選択したあとに「ビュー」パネルの「モデル」にチェックボックスを入れることで機能します）。そのため、操作中心ボーンを動かしてもモデル自身は一切動きません。

必要なボーンはありません。

### 7.2.16 ボーン表示枠に自動登録

ボーン作成時にタイムラインの表示枠に登録するかどうかを設定します。

以下のボーンは作成時に「依存するボーン」に表示枠がない場合は「センター」枠が自動的に作成され、その枠に移動します

ボーン名	依存するボーン名
腰	下半身
グループ	センター
全ての親	(最初のボーン)
操作中心	(最初のボーン)

### 7.2.17 (材質選択)

ウィンドウ右側では材質一覧が並んでおり、自動再配置の対象とする材質を選択することが可能です。例えばモデルと一体化してるアクセサリなど自動再配置の対象にしたくない材質がある場合はチェックボックスから外してく

ださい。

---

注釈: 頂点が複数の材質に紐づいている場合は自動再配置の除外設定を優先します

---

何も設定しない場合は全ての材質が自動再配置の対象となります。

## 7.2.18 変更履歴

- 1.3.0 (2021/2/14)
  - 実行後にクラッシュしやすくなる不具合を修正
  - 英語表記の文言を一部修正
- 1.2.0 (2020/12/27)
  - Apple Silicon 対応のためユニバーサルバイナリ化
- 1.1.0 (2020/8/1)
  - 必要なボーンが足りてない場合は先頭に âŽž を出すようにした
  - 実行時に必要なボーンが足りてない場合はエラーを表示して処理を中止させるようにした
    - \* これらの機能はオリジナルには存在せず独自に追加した機能となります
    - \* 29.2 においてエラーメッセージが途中で途切れる問題があります
- 1.0.0 (2020/7/20)
  - 29.2 と同時にリリース



## 第 8 章

# よくある質問と回答

---

注釈: エフェクトに関する質問は [よくある質問と回答 \(エフェクト編\)](#) で

---

nanoem を使う際において聞かれそうなこととその回答の一覧です。問題が起きて困ってる場合は [トラブルシューティング](#) を参照。

ここに書かれてることが含まれてない場合は「マシュマロ」を投げるか Twitter で [#nanoem](#) または [#nanoem 互助会](#) タグをつけてツイートすると誰かが答えてくれるかもしれません。

### 8.1 MikuMikuDance の違いはなんですか？

---

重要: nanoem は MikuMikuDance の互換実装ではありますが、**MikuMikuDance** ではありません。よって MikuMikuDance の代わりにはなりません。

互換あるいはそれに類するモーション編集が可能な実装は他にも以下があります。

- [MikuMikuMoving](#) (略称として MMM)
- [NexGiMa](#)

いずれも DirectX をベースとしているため Windows のみの対応です。

---

- nanoem は DirectX11 をつかうが、MikuMikuDance は DirectX9 をつかう
- 見た目は MikuMikuDance に似ているが細かいところの使い勝手が再現できない
- MikuMikuDance が利用するファイルを一通り読めるがエフェクト、IK 及び物理演算に完全な再現性をもたない

MikuMikuDance の単純な乗り換え先として使おうとするなら 2 番目及び 3 番めの理由による問題が大きくなるためやめたほうがよいでしょう。もし macOS で nanoem ではなく MikuMikuDance を動かしたい場合は [macOS で MMD をつかう方法](#) を参照してください。

nanoem 特有の特徴は以下です。

- DirectX11 を利用しつつも独自実装のエフェクト読み込み機能により MME との互換性を確保
  - これは MME で使われる技術は DirectX9 に依存しており DirectX11 で利用することができないためです
    - \* MMM でも DirectX11 版は独自形式のエフェクトで MME を利用することができません
  - すべての MME のエフェクトに対応しているわけではなく読み込めないものもあります
- モデル編集機能の統合
  - MikuMikuDance にも部分的に存在しますが物理演算関連に限定されます
- 高解像度モニターかつ Per-Monitor V2 によりモニター毎に DPI が違うものでも対応
- ARM64 に一応ネイティブ対応

## 8.2 モデル編集できますか？

33.0 からある程度可能になりました。ただしモデル編集におけるデファクトスタンダードであり実質唯一の実装である PMXEditor とは操作方法が大きく異なるため同じ使い勝手で利用することはできません。また PMXEditor のプラグインを利用することができません（例外として「[標準標準ボーンプラグイン](#)」はあります）。

モデル編集が出来るとはいえ PMXEditor と同じく原則として予めある程度モデルが出来た状態からの利用を想定しており、ゼロから作るのは困難を極めます。そのため以下に挙げるモデリングが可能なソフトウェアを利用して X/PMD/PMX 形式のいずれかに変換してから nanoem に持ってきてください。いずれも Windows/macOS に対応しています。

- Blender
  - mmd\_tools アドオンの導入が必要です c.f. [【Blender 2.8】MMD モデルやモーションを Blender にインポートできるアドオン「mmd\\_tools」](#)
  - mmd\_tools 導入時 Shared Toon Textures の設定で必要な共有テクスチャのフォルダはありません
    - \* これは nanoem がアプリケーションに直接共有テクスチャを組み込んでるためです
    - \* そのため MikuMikuDance をダウンロードして解凍し、その中にある Data フォルダのパスを指定してください
- Metasequoia

- PMD (not PMX) 形式で出力するには有償版が必要です
- Xismo
  - ver 2.0 以降は macOS に対応していません
  - ver 1.x でも OpenGL を利用しているため将来的には動作しなくなる可能性があります
- VRoid Studio
  - 書き出しは VRM のみで「*VRoid Studio* で作った VRM を読み込むことができますか？」で解説している通り直接 nanoem で読み込むことはできません

## 8.3 動画編集できますか？

できません。動画編集をしたい場合は以下に挙げる動画編集を目的とするソフトウェアを利用してください。

- macOS
  - iMovie
  - Final Cut Pro おしゅ
  - Adobe Premier Pro おしゅ
  - DaVinch Resolve
  - Javie
  - OpenShot Video Editor
  - Avidemux
- Windows
  - Adobe Premier Pro おしゅ
  - AviUtl
  - DaVinch Resolve
  - OpenShot Video Editor
  - Avidemux

おしゅ とついでるものは有償版のみ

## 8.4 講座動画はありますか？

ありません。また、作る予定もありません。本人曰く作るコストが見合わないことを理由としています。

## 8.5 MMD のプロジェクトファイルの読み込みはできますか？

できます。プロジェクトファイルの中身によっては不具合が起こるかもしれません。

また、書き出しも可能ですがいくつか制約があります。詳しくは「[PMM 形式の保存の制限について](#)」を確認してください。

## 8.6 VRoid Studio で作った VRM を読み込むことができますか？

できません。また、今後の実装予定もないので VRoid→MMD が楽になりましたまたは VRMtoPMXExporter を改めて使ってみました【2019 年 4 月版】にある方法で PMX に変換してください。

ご用心: 上記記事で言及されている VRM2PMXConverter は Windows 専用ですので macOS 版の場合 VRMtoPMXExporter を使う方法しかとれません (つまり Unity を使う必要があります)。

上記ツールを使っても配布モーションで事実上必須となる準標準ボーンが作成されません (変換するツール及びバージョンによっては対応している可能性があります)。その場合は nanoem 用に作られた「[準標準ボーンプラグイン](#)」で作成可能です。

## 8.7 縦長画像あるいは動画を出力したい

正しく出力するためには「ビューポートアスペクト比を維持する」機能のために 26.0 以降が必要ですが以下の手順で縦長動画を出力できます

---

注釈: 縦長は主にスマートフォン向けを想定していますが端末毎に解像度がバラバラなので解像度のプリセットとしては用意していません

---

- 「プロジェクト」 > 「ビューポート設定ダイアログを開く」
- 「出力サイズ」を出力したい解像度に変更する
- 画像または動画出力で以下の設定を行う
  - 出力解像度に「ビューポートと同じ」を選択する



- 「ビューポートアスペクト比を維持する」にチェックする
- 27.1 未満かつ動画出力の場合「OS 付属のエンコーダを有効にする」のチェックを外す
- OK ボタンを押して出力する

## 8.8 透過つき動画を出力したい

以下の手順で透過つき動画を書き出せます。

- 「プロジェクト」 > 「ビューポート設定ダイアログを開く」
- 「透過背景を有効にする」にチェックを入れる
- 動画出力時に「OS 付属のエンコーダを有効にする」のチェックが外れていることを確認
- Plugins から `plugin_ffmpeg` を選択し、以下を設定する
  - Video Codec を `Ut Video Codec` に設定
  - Video Pixel Format を `RGBA` に設定
  - Audio Codec を `PCM` に設定
- OK でファイル名を指定して書き出す

書き出された透過あり動画は

- Windows の場合は [Ut Video Codec Suite](#) を導入
- macOS の場合は macOS 向けの動画プレイヤーである [IINA](#) で再生して確認することが可能

---

重要: 上記手順でとくに以下の2項目は両方ともやらないと透過あり動画で書き出されません。

- 「透過背景を有効にする」にチェックを入れる
- Video Pixel Format を `RGBA` に設定

エフェクト（とくにポストエフェクト）についても透過出力に対応するものを使う必要があります。対応しているかどうかはエフェクトの README または取扱説明書を確認してください。

---

## 8.9 モーション調整のため最初の部分に無音を挿入したい

nanoem には無音を挿入する機能を持っていません。そのため [Audacity](#) を利用して以下の記事を参考に無音を挿入してください。

注釈: nanoem もとい MMD のモーションは 1 秒を 30 フレームで計算します。そのことから 1 フレームあたりのサンプル数は以下になります。

- 音源のサンプルレートが 44100Hz の場合は **1470** ( $= 44100 / 30$ )
- 音源のサンプルレートが 48000Hz の場合は **1600** ( $= 48000 / 30$ )

たとえば 5 フレームずらしたい場合で音源のサンプルレートが 44100Hz の場合は **7350** サンプルを指定します。

---

Audacity で曲の途中に無音部分を追加する方法！【空白/間を開ける】

## 8.10 nanoem をスリープさせるには？

macOS 版の場合は以下のどちらかをとると nanoem のスリープが発生するため CPU/GPU の消費を最小限に抑えることができます。

- nanoem のウィンドウを最小化する
- nanoem のウィンドウの上に別のアプリケーションのウィンドウをかぶせる

Windows 版の場合は nanoem のウィンドウを最小化することで同等の効果が得られます。

## 8.11 MMDAI2/VPVM とはどう違うのですか？

注釈: MMDAI2/VPVM とは 2014 年まで配布していた nanoem と同じ開発元による MMD 互換実装です。

MMDAgent を源流として macOS に対応するための実装として 2010 年 1 月に提供したものが MMDAI でそれをモーション編集できるように機能追加したものが MMDAI2 です。VPVM は MMDIA2 の基本実装を流用して派生として開発したものです。

現在はいずれも開発及び保守はすでに行われていません。

---

できることはほとんど同じですが、以下が大きな違いとしてあります。

- Metal に対応
  - MMDAI2/VPVM は設計上 OpenGL に強く依存するため Metal に対応できない
- MME に対応
  - MMDAI2/VPVM は外部ライブラリを利用しており MME とは互換性がなく、そのため専用のエフェクトファイルを作る必要があり開発負担となっていた

ほかにも MMDAI2/VPVM にはない機能として以下があり、MMDAI2/VPVM における実装上の不具合が nanoem では修正されてたりします。

- PMM 読み込みに対応
- SDEF 対応

## 8.12 あるモデルにモーションをふたつ以上適用したい

注釈: 1.15 以降は最初からマージが有効になってるため何もする必要はありません

1.14 以前の場合はデフォルトではひとつのモデルにつきひとつのモーションしか適用されませんが、「キーフレームをマージする」を有効にすることでふたつ以上のモーションを合成することが出来ます。<sup>\*3</sup>

1. 画面左パネルの「モデル」を開き、適用するモデルを選択する
2. 「モーション」をクリックし、「キーフレームをマージする」を有効にする
3. モーションを読み込む

ダンスモーションとリップモーションが別々になっている場合は上記の方法で行ってください。

## 8.13 古いバージョンがほしい

対応コストの関係で基本的に最新版しか提供していません。

どうしても古いバージョンが欲しい場合はバージョンの指定（ひとつのみ、複数不可）と古いバージョンを使いたい理由を書いた上でインストールの「不具合報告の方法」に従って送信してください。

## 8.14 アンインストールしたい

AppCleaner を使って nanoem で検索して見つかったファイルを全て削除してください。

**警告:** 単に nanoem.app をゴミ箱に捨てる方法は未削除ファイルが残るため非推奨です。

<sup>\*3</sup> 原則キーフレームがかぶらないようにすること。もし衝突した場合は先にある方が適用される

## 8.15 よろずの質問

### 8.15.1 nanoem はなんて読みますか？

日本語圏においては「なのえむ」です。以下に表をまとめます

言語	読み方
日本語 (ja_JP)	なのえむ
繁体字 (zh_TW)	納米 <sup>*1</sup>
简体字 (zh_CN)	纳米 <sup>*1</sup>
한국어 (ko_KR)	넌미

### 8.15.2 nanoem 以外の macOS で使える MMD 互換はありますか？

MikuMikuDance のように使える、となるとないです。また、おそらく今後も以下の理由からないと思われます。

- そもそも作るのが非常に難しい
  - MikuMikuDance は DirectX のエコシステムがあって成り立つが、それらに相当するものを自前で実装しなければならない
  - さらに GUI アプリケーションやその他の知識も必要で作るのに求められる知識は膨大
    - \* ゲームとゲームエンジン両方作るくらいの知識が求められます
  - ちなみに MikuMikuMoving の開発者が macOS 版の対応予定について濁してる
- macOS の利用人口が少ない
  - OS シェアとして macOS の利用人口は Windows の 1/10 以下しかない
- VRM の登場
  - 人型という制約はあるが、業界標準をベースにしたものであり執筆時点 (2018/5/27) で今後発展が進むものと思われる

ひとことでいえばハイリスク・ローリターン であるということです。

### 8.15.3 英語で対応できますか？

---

<sup>\*1</sup> 本来は「ナノメートル」を意味するのだが直接的な当て字がわからないので暫定的に相当するものをあてた。ちなみに日本語だと「年貢を納める」になり意味が異なる

---

注釈: Google Translate などの翻訳をかけて読まれることを想定してこの部分はあえて日本語で書いてます

---

できますが、原則として日本語で書いたほうが対応が早いです。ただし日本語で書くために機械翻訳を用いるくらいなら英語で書いてください。

#### 8.15.4 日本語/英語以外の対応予定はありますか？

公式に対応しているのは日本語と英語のみ<sup>\*2</sup> で現段階でないですが、日本語または英語以外の言語に翻訳出来る翻訳者がいればあるかもしれません。

もし日本語または英語以外に翻訳する翻訳者として立候補したい場合は [@shimacpyon](#) にフィードバックか DM で翻訳者希望の旨を連絡してください。

#### 8.15.5 どんな人が利用していますか？

Google Analytics にもとづく利用状況の結果から以下になります ( [プライバシーポリシー](#) 参照 )

- もともと macOS 向けに作られたアプリケーションなので、macOS 使ってる人が大半
- 約 7 割が国内 ( 日本 ) 約 3 割が国外からで最も多いのが米国

---

<sup>\*2</sup> UI 自体は英語に対応しているものの、マニュアルは日本語のみしかないという矛盾



## 第 9 章

# よくある質問と回答 (エフェクト編)

---

重要: macOS での OpenGL 利用時に発生したエフェクトの不具合はサポート対象外です (macOS 10.13 未満が明示的にレンダラを OpenGL にしている場合に該当)。Metal に切り替えてから利用してください。

---

### 9.1 エフェクトの読み込みはできますか？

できます。モデルまたはアクセサリと同時に読み込まれます。詳しくは [モデル](#) または [アクセサリ](#) を参照してください。

ポストエフェクト系やあまり複雑でないエフェクトであればだいたい読み込めますが、ハードウェア (特に GPU) の環境差異があることから MME と同じ結果になることは保証していません。

### 9.2 エフェクト詰め合わせが見当たらないのですが...

---

注釈: エフェクト詰め合わせはエフェクトプラグインを実装する前に利用していた nanoem でのみ利用可能なエフェクトのパッケージです。OpenGL 上でしか利用できず、かつバイナリ形式での提供のため改変が出来ない問題を抱えていたため現在は提供を終了しています。

---

エフェクト詰め合わせは 2020/8/31 をもって配布を終了しました。保守も行っていないため、以下のオリジナルの方を利用してください。

- otamon さんの [おこねや](#)
- sovororo さんの [エフェクト集置き場](#)

## 9.3 ray-mmd は使えますか？

できます。ただし条件付きです。

**警告：** 画像または動画として書き出すときはアンチエイリアスを無効にしてください。

**警告：** 高解像度で利用する場合は iMac Pro または 外付け GPU を利用した Mac が事実上必須です。（参考程度に MBP 13 インチ 2018 年モデル 上で外付け GPU なしかつウィンドウサイズ変更なしの状態だと 20FPS 未満）

また、低解像度に切り替えて実行する場合でも GPU のある iMac または MacBook Pro 15inch での利用が望ましいです。

---

注釈： 技術的な仕様による問題のため OpenGL 上では正常に動作しません

---

1.22.0 から暫定的に利用できるようになっていますが、そのままでは読み込むことができないのでファイルの改変が必要です。macOS の場合レンダラを Metal に切り替える必要があるので「設定」よりレンダラを Metal に切り替えてください。（Metal 非対応の場合は切り替え不可）

注意書きにあるとおり ray-mmd は特性上、性能要件が非常に高いのでお使いのマシンを確認してください。要件が外れる場合 nanoem が非常に重くなりまともに操作できなくなって再起動せざるを得なくなるおそれがあります。

### 9.3.1 改変場所

---

**重要：** ray-mmd に限らない話ですがエフェクトを改変するときは元に戻せるようにバックアップを取ってください。また、macOS の場合テキストエディタで編集すると改行部分が正しく保存されず読み込み時にエラーになってしまいます。

---

以下のファイルを Visual Studio Code のようなプログラミング用のエディタなどで FOG\_ENABLE の値を 1 から 0 に修正したあと ray-mmd の読み込みとエフェクトの割当調整を行ってください。



## リスト 1 ray.conf

```
#define FOG_ENABLE 0
```

1.22.3 未満の場合は不具合によりさらに以下の `Sky*box*` を `sky*box*` に小文字に変更する 3 箇所の追加の改修が必要です (1.22.3 以降の場合は不要です)。

## リスト 2 Shader/textures.fxsub

```
"sky*box*.* = ./Skybox/skylighting_none.fx;"
```

## 9.4 ikPolishShader は使えますか？

警告: 重さは ray-mmd と同等あるいはそれ以上です

28.0 から利用可能です。ただしいくつか注意があります

- ikPolishShader v0.26 では高品質 (2) まで対応
  - デフォルトのカスタム設定 3 はコンパイルが通らない
- ikPolishShader v0.27 では低品質 (1) まで対応
  - 普通 (2) は表示上の問題があり
  - 高品質 (3) またはデフォルトのカスタム設定 (0) は落ちる

0.26 と 0.27 とでは表示上互換性のない変更があるため両方確認しています

## 9.5 MotionBlur 系が動かないのですが...

一部の改変が必要です。これは nanoem が利用する描画バックエンドと MMD の描画バックエンド (DirectX9) のラスタライズの仕様の違いのためです。<sup>\*1</sup>

ご用心: 28.1 以前では下記にある改変を行ってもふたつ以上のモデルがある状態で MotionBlur を利用すると正しく機能しない不具合があります。この問題については 28.2 で修正されています。

<sup>\*1</sup> 技術的な話として nanoem では Draw=Buffer の場合頂点シェーダにわたす前に予めサブテクセルのズレを意図的に起こして頂点シェーダによる処理によりゼロサムにして差異を吸収する仕組みを持っていますが、モーションブラーなどで使われる頂点テクスチャフェッチのようなケースの場合は例外のため改変が必要です。

一例としてそぼろさんの MotionBlur2 の場合は `VelocityMap.fx` を編集し、「ここから追加」の行から「ここまで」の部分を追加すると正しく機能するようになります。

リスト 3 VelocityMap.fx:343

```
Out.Pos.xy = (tpos * 2 - 1) * float2(1,-1);
Out.Pos.zw = float2(0, 1);

// ここから追加
#ifdef NANOEM
Out.Pos.x += VPBufOffset;
Out.Pos.y -= VPBufOffset;
#endif
// ここまで
```

また、同じくモーションブラーを利用する `TrueCamera/TrueCameraLX` についてもファイルが `TrueCameraObject.fx` または `TCLX_Object.fxsub` で改変する行の位置に違いがありますが、改変内容は同じです。

## 9.6 '#' : `invalid directive` が出る

これは以下のようなコードを利用していると未実装のために発生します。

```
#define some_macro(n) replaced_result_##n
```

文字列結合と呼ばれる処理のため、上記の `define` の行を削除し、例えば以下のように使われている場合は

```
some_macro(test)
```

文字列を置き換えた結果を使用箇所全てに適用してください。

```
replaced_result_test
```

詳しくは [トークン連結演算子](#) を参照してください。

## 9.7 画面が固まったかのような表示になる

一部エフェクトでビューポート切替時にクリア処理がないためビューポートが固まったかのような表示になることがあります。回避策としてクリア処理の追加が必要です。

当該問題を確認している [DropShadow](#) の場合は以下の改変が必要です。

リスト 4 DropShadow.fx:212

```

"RenderColorTarget0=";
    "RenderDepthStencilTarget=";
    // ここから追加
    #if defined(NANOEM)
    "ClearSetColor=ClearColor;"
    "ClearSetDepth=ClearDepth;"
    "Clear=Color;"
    "Clear=Depth;"
    #endif
    // ここまで
    "Pass=Gaussian_Y;"

```

## 9.8 nanoem 上で動いてると MME からどう判断すればよいですか？

NANOEM マクロが予め定義されているので、その有無で判断できます。

また、nanoem では動いているレンダラにあわせて MME から変換する都合上、変換先のシェーダ形式を示すマクロが定義されています。

これらはいずれも定義された上で変換先の場合は 1 を、変換先ではない場合は 0 を示す値が入ってるため、定義の有無だけではなく数値の値も判断する必要があります。

マクロ名	対応するレンダラ	変換先のシェーダ形式（言語）
NANOEM_OUTPUT_SHADER_LANGUAGE_GLSL	OpenGL	GLSL
NANOEM_OUTPUT_SHADER_LANGUAGE_ESSL	OpenGL (ES)	GLSL
NANOEM_OUTPUT_SHADER_LANGUAGE_HLSL	DirectX	HLSL
NANOEM_OUTPUT_SHADER_LANGUAGE_MSL	Metal	MSL
NANOEM_OUTPUT_SHADER_LANGUAGE_SPIRV	(将来予約用)	SPIR-V



## 第 10 章

# トラブルシューティング

nanoem で利用する際に問題が発生したときの回答一覧です。

### 10.1 nanoem が開かない、起動しない

---

**重要:** macOS 10.12 を利用している場合は起動直後に落ちる問題を確認しているため可能であれば macOS 10.13 以降にアップグレードしてください。なお、この問題は 29.1 以降で修正しています。

---

起動時に警告ダイアログが出る場合は「Mac で App を安全に開く」の「ノータリゼーションを受けていない App や未確認の開発元の App を開く方法」を確認してください。

ダイアログなしに起動しない場合は起動時に必要なフォルダのアクセス権限が不適切なことが原因の可能性があります。詳細は以下を参照してください。

- Mac OSX で急にソフトが起動しなくなったときはアクセス権を疑ってみよう
- ファイルやアプリケーションが突然開けなくなった！

それでも解消しない場合は使っている macOS のマイナーバージョンが古いことが原因の可能性があります。ソフトウェア・アップデートより macOS のマイナーバージョンの更新を行ってください。

- Mac のソフトウェアをアップデートする

### 10.2 音源が読み込めないあるいは音がでない

---

**重要:** 31.5.0 において WAV 以外の音源が読み込めない場合がある不具合の修正及び音源読み込み時のエラー検出の強化を行いました。

エラーが出ない場合は「[Mac のスピーカーの音が聞こえない場合](#)」を確認してください。また、ミュートになっていないかの確認を行ってください。

---

音源が読み込めないケースで macOS 版の場合

- iTunes か QuickTime で読み込むことができるかを確認してください
  - それらで読み込めない場合は nanoem でも読み込めないので音源を確認してください
- iTunes または QuickTime で読み込めるが nanoem で読み込めない場合は一回 wav (PCM) 形式に変換してから nanoem に読み込ませてください

音源が読み込めてるが音源全体で音が出ないケースの場合は

- 他のアプリでは音が出るが nanoem で音が出ない場合は [再生パネル](#) の音量が 0% になっていないことを確認した上で nanoem を再起動してください
- nanoem 含む他のアプリでも音が出ない場合は Windows/macOS を再起動してください

## 10.3 謎の爆音が発生することがある

macOS において起動してから数時間が経過しかつ音源が読み込まれた状態で爆音が発生することがある現象が報告されていますが、2021/7/14 現在でも原因がわからないため長時間作業が想定される場合は以下で対応してください。

- 「[音源を消去](#)」を行う
- nanoem を再起動する

## 10.4 背景動画が読み込めない

macOS 版においては QuickTime で読み込み可能かどうかを確認してください。QuickTime 以外の場合は [VLC](#) あるいは [IINA](#) で確認します。QuickTime/VLC/IINA いずれも読めない場合はそもそも動画として正しく認識できていない可能性が高いです。

読み込むとクラッシュする場合は avi 形式に変換することで内部で使われる実装が切り替わるため読み込める可能性があります。

## 10.5 プロジェクトが読めなくなってしまったのですが

**重要:** 31.5.0 以降 nmm 形式においてプロジェクトファイル読み込み時に部分的にファイルが読み込まれなかった場合は何が読み込まれなかったのかをダイアログ表示するようにしました。

25.0.0 以降では音源または動画のファイルが見つからない場合はエラーではなく単純に読み込みを無視するようになりました。そのため、音源または動画が読み込まれない場合は元のファイルが何らかの理由で見つからないことが原因の可能性があります。

---

[bowlroll](#) または何らかのアップローダを使ってプロジェクトファイル一式をアップロードし、本人と開発者以外からみて秘匿出来る形<sup>\*1</sup>で[不具合報告の方法](#)で報告してください。可能な限り対処しますが、プロジェクトファイルの状態によってはだめな場合があります。

## 10.6 プロジェクトが保存出来ないのですが

---

**重要:**

- ボーンまたはモーフを動かした状態でキーフレーム未登録が残ったままプロジェクトを保存しようするとエラーが出ます。
  - 31.0 まで既存のモーションを読み込んで保存しようすると場合によってエラーで保存できない、もしくは最初のキーフレームしか保存されない不具合がありました。当該不具合は 31.1 で修正しています。
  - 26.0 までモデルまたはアクセサリを削除してから保存すると必ず失敗する不具合がありました。当該不具合は 26.1 で修正していますが、もし 26.1 未満で継続利用する場合はモデルまたはアクセサリの削除をしないようにお願いします。
- 

「保存先に異常がないかを確認してください」のエラーが出た場合は以下の原因の可能性があります

- 上記の「重要」のところに引っかかてないか
- 保存先がのストレージ容量が足りていないか
- 読み込み専用に対して保存している

2 番めの場合は [GrandPerspective](#) などを利用して不要なファイルを削除し、ストレージ容量を確保してください。

3 番目の場合は保存先を変更する（例えばデスクトップなど）ことによって保存できる可能性があります。

これらでも対処できない場合は macOS の場合は「Mac の問題を診断する」で Mac 本体に異常がないかを確認してください。

---

<sup>\*1</sup> BowlRoll つかってるなら「指定ユーザのみアクセス許可」で ID:145 を指定すると確実です。c.f. [投稿者のための BowlRoll の使い方まとめ](#)

上記以外の場合は使ってるバージョンが最新版かどうかを確認し、こういった問題が発生して保存できてないのかを書いた上で[不具合報告の方法](#)で報告をお願いします。

## 10.7 モデルが正しく表示されない

---

重要: 33.0 から「[モデルデータの検証を実行](#)」が追加されました。表示トラブルが発生している場合は実行して確認してください。

---

もし全部あるいは部分的に白く表示されている場合はモデルのテクスチャが正しく読み込まれてない可能性があります。この場合は一回問題が発生しているモデルを退避し、必要に応じてモデルのダウンロードを行い zip などの圧縮ファイルを展開しなおしてください。

上記以外の部分的に正しく表示されていない場合は [不具合報告の方法](#) にて報告をお願いします。原因が多岐に渡るため、一概にいえる解決策がないためです。

また、もし以下のような現象が発生している場合

- モデル読み込むとボーンしか表示されない
- 画面が真っ黒

描画エラーの可能性が疑われますが根本的な原因不明で、解決策がない状態です。応急的な対処として以下の順で試してください。

- nanoem を再起動する
- 上でダメなら Mac を再起動する
  - 起動後に常駐するアプリケーションも重要なもの以外を落とすこと
  - 使用するとき nanoem 以外のアプリケーションを立ち上げない

## 10.8 モデルを読み込んだら落ちる

「設定」の「全体設定」にある「モデル描画の負荷分散を有効にする」のチェックを外してモデルを読み込んでください。

それでも落ちる場合は落ちたモデルの情報提供（配布元が分かる場合その場所をのせる）とともに[不具合報告の方法](#)で報告をお願いします。



## 10.9 モデルの足の部分がガクガクする

IK の仕様です。特にモーションの作成元モデルと読み込み対象のモデルの身長差が大きいと意図した位置を求めることができずより顕著になるため以下の方法で対処してください。

- 「編集」の「補正ダイアログを開く *IKI*」で補正する
- 「モデルパネル」から対象の IK ボーンを選択し *Off* をクリックして「登録」を押す
  - 登録後ガクガクしている IK リンクボーンを改めて回転し直してボーンパネルからキーフレーム登録
    - \* IK リンクボーンはボーン選択画面において黄色で表示されます
    - \* IK ボーンは Off の状態において動かしても何も機能しないため動かす必要はありません

## 10.10 アクセサリを読み込んでも何も表示されない

アクセサリファイルを開いて中身が 1 行に全て詰め込まれている場合は実装都合上 nanoem で正しく解析が行うことが出来ないため読み込むことが出来ても何も表示されません。

対策として Metasequoia を利用してファイルを一回取り込んで別のファイル名で保存するか、あるいは [homebrew](#) などで [assimp](#) をインストールして以下のコマンドラインで変換を行ってください。

ご用心: Blender ではアドオン経由での書き出ししか対応しておらず、加えて 2.80 以降に対応していません

```
# homebrew の場合
brew install assimp

# input.x は nanoem で読み込めないアクセサリファイルのパス
# output.x は書き出し先、ファイル名は任意
assimp export /path/to/input.x /path/to/output.x
```

## 10.11 アクセサリまたはエフェクトを読み込んだら落ちる

落ちたアクセサリまたはエフェクトの情報提供（配布元が分かる場合その場所をのせる）とともに[不具合報告の方法](#)で報告をお願いします。

## 10.12 新規プロジェクト作成のたびにエラーが表示される

オーディオ出力の初期化に失敗している可能性があります。他のアプリケーションで音が鳴るかを確認してください。なお、エラーが出て再生は可能ですが音源を読み込んでも鳴りません。

## 10.13 動画出力を行うと落ちる

32.0 未満の場合はアンチエイリアスを有効にすると落ちる問題が確認されています。もし 32.0 未満の場合は 32.0 にアップデートするか、アンチエイリアスを無効にして書き出してください。

32.0 以上の場合でも場合によってはアンチエイリアスを有効にした状態で落ちることがあります。その場合は別途アンチエイリアスエフェクトを利用する形でアンチエイリアスを無効にして書き出してください。

## 10.14 動画出力で書き出すと意図しない（例えば真っ黒）動画が出力される

もし「OS 付属のエンコーダを有効にする」にチェックをつけて出力していた場合は無効にし、かわりに Plugins の欄から `plugin_lsmash` を選択して動画出力を行ってください。

レンダラ が OpenGL の場合でかつ利用可能な場合は Metal に切り替えることによって解決することがあります。

それでも解決しない場合は macOS のバージョンをあげてダメなら nanoem での動画出力を諦めて QuickTime Player の画面収録を使うしかありません。

## 10.15 動画出力で書き出すと再生では起こらなかったフレーム欠けが起こる

---

重要: 31.5 から動画出力設定画面に「フレームズレ抑止を有効にする」が追加されました。もし無効になっている場合は有効にしてください

---

「OS 付属のエンコーダを有効にする」を有効にすると発生することまでは確認できています。が、技術的仕様上原因不明で修正できない状況のため、以下の方法で対処してください。

- [新規プロジェクト作成のたびにエラーが表示される](#) にある方法で対処する
- 動画編集でフレーム欠けを起こしてる箇所を取り除いて前後のフレームから補完する

## 10.16 ウィンドウ画面が水色に表示される

レンダラに OpenGL を利用している場合 nanoem 側の不具合によって発生します。もし遭遇した場合は[不具合報告の方法](#)で報告をお願いします。

レンダラが Metal だと発生しないため報告しない限り放置される可能性が高いです。

## 10.17 カメラモーションを読み込んだら角度の X 軸が反転する

---

重要: 33.0 で修正済みです。32.0 以前に保存したプロジェクトは読み込み時に補正します。

---

32.0 以前の不具合によるものです。「編集」>「カメラ」>「補正ダイアログを開く」の「角度」の最初の項目の \* 1.00 を \* -1.00 に変更し、「OK」で実行してください。

## 10.18 物理演算で貫通を起こす

物理演算部分は現在も調整中のため貫通（とくにスカート）が起こりやすいです。物理演算の仕様もあり一律で完全に防ぐことが不可能のため、以下を試してください。

- カメラで貫通部分を隠す
  - 一番簡単な方法だが見せられる部分が減る
- 剛体またはジョイントのパラメータを調整する
  - 設定画面からモデル編集機能有効にしたあとメニューの「編集」の「モデル編集ウィンドウ」を開く  
と出る「剛体」と「ジョイント」のパラメータは編集可能です
  - ただし操作と調整が非常に難しいためどうしてもという時のみ使ってください
- ボーンパネルの「物理」のチェックボックスを外して登録する
- 「[ボーン単位で物理演算を無効化する](#)」を参照
- 26.0 以降から実装された機能で「物理」のチェックボックスを外して登録したキーフレームは物理演算が無効になります
- 有効と無効の間の補完仕様があるため扱いが難しいですが、物理演算の切り替えを制御できる唯一の手段となります

## 10.19 カメラを動かすとセルフシャドウが消えることがある

---

重要: 29.0 未満までセルフシャドウの計算方法に問題があり、これを修正したため頻度は軽減しています。

---

セルフシャドウの仕組みの限界による仕様で、不具合ではありません。発生した場合はモードを切り替えるかセルフシャドウの有効距離を調整してセルフシャドウのキーフレームに登録してください。

外付け GPU を搭載している Mac をお使いの場合は ExcellentShadow などの上位なセルフシャドウ用エフェクトを利用する手もあります。完全になくすことはできませんが、発生しにくくなります。エフェクトの README にもありますが適用すると動作が重くなります。

## 10.20 zip を解凍したらファイル名が文字化けしたのですが

The Unarchiver などのアプリを使って解凍してください。これは zip 内のファイル名の文字コードを日本語版 Windows 以外では正しく認識できないことが原因です。

MMD における zip はほとんどが日本語版 Windows で作られてるので日本語版の Windows では問題ないのですが、日本語以外の Windows 及び macOS など扱う場合は上記の理由により文字化け問題を引き起こします。

## 10.21 タイムラインとビューポートの分離はどうやったらできるの？

33.0 以降から「[ビューポートウィンドウを分離](#)」で分離することができます。それより前のバージョンはタイムラインとビューポートの比率のサイズ調整のみ可能です。

ただし MMD のように独立したウィンドウとして分離するのではなく、あくまでアプリケーション内のウィンドウとして分離される形のためウィンドウを外に持っていくことはできません。

## 10.22 外部親設定したら意図しないモデルに紐付いた

外部親の対象モデルは名前で紐づく仕様のため、モデル名がプロジェクト内で重複する場合（同じモデルで複数回読み込むと発生します）は最初のモデルが外部親に設定されます。

モデル名が重複している場合は「[（モデルの選択）](#)」を参照の上で設定対象以外のモデル名を変更してください。

## 10.23 一時停止を繰り返すと再生時にかくつくんですが...

---

注釈: 音源を読み込まなくても当該問題が発生します。これは内部的に無音の音源が読み込まれるためです。

---

音源同期補正処理による影響です。一時停止を繰り返すと音源のズレが大きくなり音源同期補正処理が毎回強制的に働くためです。

対処方法として画面右下の一時停止ではなくメニューのほうの停止を選択してください。補正処理がリセットされるためかくつきが発生しなくなります。ただし停止後に一時停止を繰り返すと再発します。また、音源同期補正処理を無効にする方法は提供していません。

## 10.24 一部文字が「？」と出て表示されない

nanoem では組み込みフォントとして源実フォントのひとつである **源実ゴシック P** を利用しているため、当該フォントに収録されていない文字は表示できません。これについては仕様です。

また内部処理の関係上、ASCII および日本語以外の文字の表示は対応していません。

## 10.25 VRM から変換したモデルを読み込ませるとボーン表示のみになる

24.x 系以前に起きていた VRM 変換時に付属するシェーダを起因とする不具合によるものです。以下の方法で対処してください。

- 25.0.0 以降を利用する
- (24.x 系以前を使い続ける場合) 付属シェーダを削除するか書き換える
  - 書き換える場合は *half2 half3 half4* をそれぞれ *float2 float3 float4* に書き換える
  - どちらにせよトゥーン表示が間違っただけのため意図した色にならない問題が残る
    - \* トゥーン表示が間違ってる問題は 25.0.0 以降で対策しています

## 10.26 macOS 10.15 以降で「キーボード入力監視」の確認を求めるダイアログが表示される

macOS 10.15 Catalina で導入された仕様のため可能であれば当該問題に対処した 25.0 以降を利用してください。トラックパッド有無の検知で利用してるため無効にした場合でも動作に支障をきたす問題はありません。

## 10.27 OSStatus returns 560558962 が出る

---

注釈: 24.0 の不具合で WAV 以外の音源を読み込むと発生してました。24.1 で修正してます

---

nanoem が利用している macOS の AVFoundation のエラーです。

音源を WAV(PCM) に変換してから読み込んでください。また、発生原因である音源の周波数（サンプリングレート）が 8KHz から 192KHz 以内におさまってるかを確認してください。それでもなおらない場合は [不具合報告の方法](#) にある手順で報告をお願いします。

## 10.28 OSStatus returns 2003332927 が出る

上記に同じく AVFoundation のエラーですが、こちらは発生原因が不明のため[不具合報告の方法](#) にある手順で報告をお願いします。

## 10.29 This effect cannot be compiled due to the renderer is not OpenGL が出る

警告:

- エフェクト詰め合わせは配布を終了しました
- macOS では OpenGL が非推奨のため将来的に動かなくなる可能性があります

これはレンダラが OpenGL 以外の環境でかつ過去に配布されていた「エフェクト詰め合わせ」に含まれるエフェクトを使用していると発生します。対処法については [エフェクト詰め合わせが見当たらないのですが...](#) を参照してください。

## 第 11 章

# プライバシーポリシー

nanoem ではふたつ (32.x 以降はひとつ) の情報が送信されます。その利用目的を理解した上で使用してください。

### 11.1 強制終了時の情報送信

nanoem が何らかの原因で強制終了すると [Sentry](#) を通じて強制終了時の情報が送信されます。

---

ちなみに: オプトアウト式をとっており、「設定」>「Global」>「クラッシュレポートを有効にする」のチェックを外すことで無効化出来ます。

---

---

注釈: 26.0 から Sentry を利用するように変更されました。26.0 より前は [AppCenter](#) を利用していました。26.0 より前において強制終了時の情報送信は無効化出来ません。また、強制終了後の起動時にダイアログが出ますがそちらに連絡先を書いても無意味でこちらから連絡することはないため記入しないでください。

---

#### 11.1.1 目的

nanoem の利用状況及び強制終了時の原因特定と不具合修正のために送信されます。強制終了時の情報は秘匿情報として取り扱います。

### 11.2 利用状況の情報送信

---

重要: 32.0.0 から Sentry に一本化するため Google Analytics (Google Measurement Protocol) 利用を廃止しました

---

1.15.0.0 から [Google Measurement Protocol](#) によるトラッキングを追加しています。

---

ちなみに: オプトアウト式をとっており、「設定」>「Global」>「アクセス解析を有効にする」のチェックを外すことで無効化出来ます。

---

### 11.2.1 目的

nanoem の利用状況の可視化を目的として匿名のデータとして送信されます。利用状況は随時公開されることがあります。



## 第 12 章

# 変更履歴

---

注釈: 29.2.0 以前の変更履歴は `change_log_past` を参照してください

---

### 12.1 33.4.0 (未リリース)

#### 12.1.1 機能追加

- 「モデルの身長測定」の追加

#### 12.1.2 仕様変更

- モデルモーション読み込み後にモデルに含まれるボーンまたはモーフに該当するキーフレームを全選択するようにした
  - MMD と挙動をあわせるための仕様変更となります
- キーフレーム登録時に登録されたキーフレームも同時に選択されるようにした
  - 上と同じ理由です
  - 処理の都合上クラッシュリカバリにはキーフレーム登録時の選択状態が保存されません
- nmm 及び pmm 形式においてキーフレーム選択状態を考慮する形で読み込み及び保存するようにした
  - nmm の場合は 33.4.0 未満から読み込んだ初回のみ従来どおり未選択状態になります
  - 33.4.0 以降で nmm を保存して読み込んだ場合はキーフレームの選択状態が復元されます

### 12.1.3 不具合修正

- 33.0 以降で背景動画が読み込まれるもののビューポートに表示されない
- pmm 形式での保存時に保存処理でフリーズしてしまい強制終了以外に終了させることができなくなることがある
- モデル読み込み確認などで利用されるダイアログ表示中に各種操作が実行可能で条件次第ではそれ起因で落ちる
  - ダイアログ表示中に保存処理が行われた場合は進捗画面が表示されますが何もしません（本来行われてはいけないため）
- エフェクト利用時の画像のミップマップ生成数のミスにより落ちることがある
- `plugin_lsmash` を使って動画出力時にキャンセルすると強制終了以外に終了させることができなくなることがある

## 12.2 33.3.0 (2021/8/31)

### 12.2.1 不具合修正

- 音源を一回削除して別のファイルを読み込もうとすると最初に読み込まれた音源が再利用されることがある
- WAV 音源が正しく読み込まれない

## 12.3 33.2.0 (2021/7/19)

### 12.3.1 不具合修正

- カメラキーフレームを登録すると X の角度が反転する
- macOS において爆音が発生する現象を抑制する予防策を追加
- エフェクト読み込み時に例外処理すり抜けが原因でエラー表示が出来ずに落ちることがある

## 12.4 33.1.0 (2021/6/26)

### 12.4.1 不具合修正

- macOS 10.14 未満で利用すると落ちる

- ボーンをカーソルに当てたときのツールチップが表示されない
- 特定のアクセサリファイルを読み込むと落ちる

## 12.5 33.0.0 (2021/6/10)

### 12.5.1 機能追加

- モデル編集機能を大幅強化
  - 利用可能な機能については model を参照してください
- モデル上に表示されるハンドル操作時に選択中のボーンの接続情報を表示するようにした
- ビューポート右下に表示されるハンドル操作時に選択中のボーンの操作軸を表示するようにした
- 以下のメニューアイテムを追加（下 2 つはモデル編集用途のためデフォルトでは表示されない）
  - モデルデータの検証を実行
  - ビューポートウィンドウを分離
  - 新規モデル
  - 現在選択中のアクセサリをモデルに変換
- Grass Valley HQX Codec でエンコードされた動画を 背景動画 として利用できるようにした
  - 背景動画として使えるようにするのみで Grass Valley HQX Codec での動画書き出しはできません。また書き出し対応の予定もありません
  - ffmpeg による互換実装のデコーダのため動画によっては正しく読み込めない場合があります
- 「選択」において Alt + Shift を押しながらドラッグすると回転する機能を追加
  - 実験的な機能のため予告なく仕様変更または削除する可能性があります

### 12.5.2 仕様変更

- モデル上に表示される移動または回転ハンドルを操作中選択されていない軸を半透明にする処理を追加
- 「右腕」または「左腕」を親に持つボーンに対してローカル軸を自動的に設定するようにした
- 選択されたボーンが移動または回転不可の場合ビューポート下のパラメータ入力のそれぞれの項目を無効にするようにした

- 高解像度ビューポートまたはアンチエイリアス有効時にビューポートに表示されるボーン接続やハンドルなどのアンチエイリアスを有効にするようにした
  - 変更前は表示負荷対策のため一律で無効にしていました
- 設定画面などの子ウィンドウの移動をタイトル部分のドラッグのみに制限するようにした
  - 変更前は子ウィンドウ内の任意の箇所にドラッグすると移動できる状態でした
- 非表示のモデルに対して物理演算を発生させないようにした
  - MMD の場合はモデルが非表示であっても物理演算の衝突判定が発生するのでご注意ください
- モーションファイルの読み込み時にカメラ及び照明のモーションかモデルモーションかをチェックするようにした
  - 従来は単純に無視していましたがそれゆえ間違って読み込んだ場合に認識できない問題があるためエラーを出すようにしました
  - 古い nanoem でカメラ及び照明モーションを書き出したファイルには対象名が設定されていないためエラーが出る可能性があります
- モデルモーションを読み込むときモデルが選択されていない場合はエラーを出すようにした
- モデルモーション読み込み時に対応するボーンあるいはモーフが見つからなかった場合は警告を出すようにした

### 12.5.3 不具合修正

- モデル上に表示される移動または回転ハンドルを操作すると位置が飛ぶことがある
- ビューポート設定変更でマウスカーソルの位置ズレを起こすことがある
- Windows 版で DirectX 利用時に対応しない MSAA を指定すると落ちる
- Windows 版において DPI の異なるディスプレイに移動させたときスケールがおかしくなる
- プロジェクトが相対パスで保存されているにもかかわらず設定上では絶対パスが指定される
- カメラのズームインあるいはズームアウトするとモーフの変形状態がリセットされる
- 戻す方向にシークすると材質モーフがリセットされる
- カメラ原点から遠く離れた場所からリセットする時にモデルのエッジ表示が異常に太くなる
- エフェクト設定画面を開いてない状態でエフェクトファイルを割り当てると落ちる
  - エフェクト設定画面を開くように指示するエラーを出すように変更しました

- 非パースペクティブ時のカメラ計算が不正なことが原因で非パースペクティブを選ぶと何も描画されていないように見える
- `shared` キーワードを利用したエフェクトにおいてオフスクリーン内の描画対象が描画されないケースがある
  - 具体的には `msToonCoordinator` で問題が顕在化
- モデル読み込み時にテクスチャが格納されているフォルダがモデルフォルダの外側にあると白く表示される
- プロジェクト保存時にモデルが未選択の場合読み込み時に未選択状態が反映されない
- Windows 版でエラー表示において文字化けが発生することがある
- `Post_ScreenTex.fx` を利用すると描画が正しく行われない
- カメラモーションを読み込んだときのアングルが間違っている（X 軸が反転している）
  - 32.0 以前で保存されたプロジェクトについては読み込み時に自動的に補正します
  - カメラモーションとして書き出してそれを読み込んだ場合は補正しないため元のプロジェクトから再度書き出してください
- プロジェクト読み込み失敗時及びエフェクトのメモリリークが発生していた問題
- 24bit 音源読み込み時に波形表示が行われない
- 背景動画に音源が含まれている場合正しくデコードされない場合がある
- エフェクト利用時に背景動画が正しく表示されない場合がある
- Windows 版において背景動画つきでプロジェクト保存した後それを読み込んだ場合背景動画が表示されない
- Windows 版において画像あるいは動画書き出しの時プロジェクト変更があったときの保存確認ダイアログが表示されない

## 12.6 32.0.0 (2021/3/8)

### 12.6.1 機能追加

- ソフトボディ実験的対応

### 12.6.2 仕様変更

- nanoem アプリにおける Google Analytics (Google Measurement Protocol) の利用廃止

- 詳細は [プライバシーポリシー](#) にて
- Sentry によるクラッシュレポート機能は引き続き使われます

### 12.6.3 不具合修正

- 画像あるいは動画出力時にポストエフェクトが含まれる状態でアンチエイリアスを有効にして実行すると落ちることがある
- アンチエイリアス設定後にエフェクトの RENDERCOLORTARGET セマンティックのテクスチャに対してアンチエイリアスが適用されない
  - 画像あるいは動画書き出しにおいてアンチエイリアスを有効にしてもポストエフェクトに対するアンチエイリアスが行われてなかった
  - オフスクリーンテクスチャはエフェクト側で制御する仕組みのため影響を受けない
- 画像あるいは動画出力画面時に出るダイアログを一回キャンセルし、もう一度行って破棄を押すとダイアログが二重に出てしまい落ちることがある
- macOS 版において Metal 利用時に「OS 付属のエンコーダを有効にする」にチェックを入れた状態で書き出すと真っ黒な動画が生成される
- Windows 版でメニューのアクセスキー（ニーモニック）が正しく機能しない
- Windows 版でファイル保存時に拡張子が二重になってしまうことがある

## 12.7 31.5.0 (2021/2/22)

### 12.7.1 機能追加

- nmm 形式のプロジェクトファイル読み込み時にアクセサリまたはモデルの読み込みが失敗した時何が原因で読めなかったか表示するようにした
- [動画（書き出し）](#) に「フレームズレ抑止を有効にする」を追加
  - 31.2.0 で導入した書き出し高速化で垂直同期の速度を上回ると動画書き出し結果が不正になる問題があったため

### 12.7.2 不具合修正

- 言語設定を英語にした時反転ペーストが機能しない
- [演算設定](#) が実質的に機能しない

- `plugin_avfoundation` において音源のデコードに失敗することがある
  - macOS において WAV 音源以外の読み込みに失敗して原因になった可能性が高い
- 接続先剛体がないジョイントが含まれたモデルを読み込ませると落ちる
  - ただしそのモデルは MMD で読み込ませると確実に落ちるので注意が必要です

## 12.8 31.4.0 (2021/1/21)

### 12.8.1 不具合修正

- 物理演算の焼きこみを行うと落ちる
  - 内部的にはモデルキーフレームにおける物理演算の有効無効の切り替え時に落ちる問題があった
- キーフレームを選択した状態かつ補完曲線表示した状態で巻き戻しを実行すると落ちる
- 言語変更時にアンチエイリアスメニューの消失およびメニューのチェック状態が正しくリストアされない
- メニューの地面影が本来有効なのでチェックされてるべきなのにチェックされていない
- macOS の OS 付属エンコーダを利用して書き出した結果が不正になる
  - 動画書き出しで落ちやすくなる問題をある程度修正して安定性を向上させた

## 12.9 31.3.0 (2021/1/15)

### 12.9.1 仕様変更

- モーフ変形時キーフレームに未登録のボーンの変形状態を含めて保存するようにした
  - いままではモーフを変形させるとボーンの変形状態をリセットする仕様だった
- PMX 2.1 で定義されているフリップモーフ及びインパルスモーフを暫定的に実装
- 次バージョンでのソフトボディ対応のためソフトボディ対応の物理演算実装に切り替え
  - 上記により物理演算で動作が変わる可能性があるかもしれません
  - ソフトボディのあるモデルの読み込みは 31.3.0 から対応していますが動作はしません

### 12.9.2 不具合修正

- 頂点が存在しないモデルにおいてモーフ処理が行われない

- nmm 形式のプロジェクト保存時に物理演算の無効及びトレースモード以外の処理モードを選択した場合保存されない
- macOS 版上でプロジェクト保存したときファイルパスに濁点半濁点が含まれると macOS 以外でプロジェクトを開こうとしたときアクセサリまたはモデルが読み込めない
  - macOS 上でしか使わない場合は対応不要です
  - もし macOS 以外で使う場合は影響を受けない macOS 版を利用して以下の対応を行ってください
    - \* 該当するアクセサリまたはモデルのモーションを一回書き出して 31.3 以降で読み込みし直した上で保存してください
    - \* アクセサリまたはモデルのファイル名だけでなくフォルダ名も対象となります
- macOS 版において OS 付属のエンコーダ利用時に書き出しすると落ちることがある
- 特定のテクスチャの不正デコードによりそれを利用するテクスチャのアクセサリまたはモデル描画に意図しない描画が発生する
- プロジェクト再生終了時の処理漏れによりアクセサリ、カメラ、光源、セルフシャドウが正しくリセットされない
- Windows 版で初回起動時に落ちる不具合が別要因で残ってた
- Windows 版で再生がかくつく
  - 再生と一時停止を繰り返すと音源の同期補正処理の関係でかくつく問題があります
  - 上記問題は一時停止ではなく停止を行うことにより同期補正処理がリセットされるため、かくつきがなくなります
- 内部的な安定性向上のための対応（落ちる箇所を特定できた箇所について修正）

## 12.10 31.2.0 (2021/1/3)

### 12.10.1 仕様変更

- 動画書き出しの処理の見直しにより処理速度を改善

### 12.10.2 不具合修正

- macOS 版において Metal で動作時ウィンドウをリサイズするとレイアウトが崩れる
- macOS 版において Metal で動作時にハングして CPU が高負荷状態のまま反応しないことがある



- プロジェクトの新規作成、プロジェクトの開き直し、再生もしくは編集時の FPS 変更において高確率に発生する

## 12.11 31.1.0 (2020/12/31)

### 12.11.1 仕様変更

- macOS 版において以下の拡張子を nanoem で直接開けるように拡張
  - nma
  - nmd
  - nmm
  - pmd
  - pmm
  - pmx
  - vmd

### 12.11.2 不具合修正

- 既存のモーション流し込みでプロジェクト保存したとき不完全な状態で保存されることがある
  - プロジェクト保存時「不完全なデータが保存されようとしたため保存が中断されました」エラーの原因と同一の可能性あり
- 背景動画が正しく描画されない
  - 31.0.0 の描画処理変更による影響が原因
- 内部変更により TEXTUREVALUE を利用したエフェクトの処理速度を改善
- モデルごとにレンダーターゲットを必要とするエフェクトにおいてメモリ使用量が激増する
- Metal 上で点描画を利用するエフェクトにおいて異常描画が発生する

## 12.12 31.0.2 (2020/12/25)

### 12.12.1 不具合修正

- Windows 版で起動直後に落ちる

## 12.13 31.0.1 (2020/12/25)

### 12.13.1 不具合修正

- macOS 版でかつ OpenGL 上での動作時ウィンドウをリサイズしようとするすると落ちる

## 12.14 31.0.0 (2020/12/25)

### 12.14.1 機能追加

- Apple Silicon Mac の暫定対応
  - 描画処理の全面的な見直しのため Apple Silicon 以外でもパフォーマンスが改善する可能性があります
  - nanoem > Preferences... > 「システム情報」の CPU が ARM (64-bit) であれば Apple Silicon ネイティブで動作しています
- PMM ファイルの書き出しを実装
  - 注意事項については [PMM 形式の保存の制限について](#) を参照してください
- PerMonitor v2 の対応
  - DPI の異なるディスプレイへの切り替えでメニューのフォントサイズが正しく追従するようになります
  - Windows 版のみ

### 12.14.2 仕様変更

- 内部で利用しているグラフィックスライブラリの制限値の引き上げ
  - 31.0 未満を利用している場合は自動的に引き上げされませんので nanoem > Preferences... > Special にある Initialize ボタンを押してください

- 通常利用で引っかかることはまずありませんが、エフェクトを多用している場合はこれ起因で落ちることがあるため上記による引き上げを推奨します
- Visual Studio 2015 から Visual Studio 2017 のランタイムに切り替え
  - Windows 版のみ
- ミップマップ生成をデフォルトで無効に変更
  - 現状の処理が潜在的に落ちる原因を作ってしまうため

### 12.14.3 不具合修正

- オフスクリーン Main における個々のモデルまたはアクセサリの表示切り替え及び消去が正しく動作しない
- 解像度倍率の異なるディスプレイへの移動時に倍率追従が正しく行われない
  - Windows 版において PerMonitor v2 対応と同時に修正していますが macOS 版も同じように修正しています
- アクセサリ削除時頂点及びインデックスバッファが削除されずメモリ上に残り続ける
- エフェクトのパラメータ画面で特定条件下で落ちる

## 12.15 30.3.0 (2020/10/27)

### 12.15.1 機能追加

- emd を読み書きする機能を追加
  - ファイルメニューではなく effect から読み込みする方式です
- モデルまたはアクセサリ名を変更する機能を追加
  - 「(モデルの選択)」及び「(アクセサリの選択)」を参照してください

### 12.15.2 不具合修正

- IK 無効化時の IK リンクボーンの無効化処理が正しく行われない
  - 結果として 全てのモデルのモーションを焼き込む (IK 含む)  $\dot{aRt}$  の結果が意図しないものになっていた
- 演算設定 及び 全てのモデルのモーションを焼き込む (IK 含む)  $\dot{aRt}$  でモデルモーションの書き出しができない

- 別モデルを選択してから再度書き出したいモデルを選択することによる回避は可能
- エフェクト利用時 APNG のデコードに失敗すると落ちる
- nmm 形式のプロジェクトファイル読み込み時にモデル名がプロジェクト内に重複していると先にモデルが適用され後からのモデルが読み込めない
- IK 有効時 IK ボーンを動かすと IK リンクが未登録判定される
- 付与親のもつ付与回転または付与移動が適用されない

## 12.16 30.2.0 (2020/9/21)

### 12.16.1 機能追加

- 音源及び背景動画を削除してリセットする機能を追加

### 12.16.2 不具合修正

- エフェクト利用時モデル及びアクセサリの加算が効かない
- エラー発生時特定条件下で落ちる
  - エフェクトまたは標準ボーンプラグイン利用時に発生しやすいが原理的にはそれ以外でも発生する可能性があった

## 12.17 30.1.0 (2020/8/31)

### 12.17.1 機能追加

- WAV (PCM) 音源をドラッグアンドドロップで読み込めるようにした
  - プラグインが必要な音源は読み込めないため従来どおりファイルダイアログから読み込んでください

### 12.17.2 不具合修正

- テクスチャ名指定がファイルではなくフォルダになっている場合エラーが発生して読み込めない
  - フォルダ指定の場合テクスチャが存在しないものとして扱い読み込みを続けるようにしました
- ボーンまたはモーフ操作中に不意にカメラ操作が発動しないようにする仕組みを導入

- エフェクトの `#include` 句において階層構造が含まれるときにレアケースで機能しないことがある
- カリング無効と有効の材質が両方あると先に出た片方しか反映されない
- プロジェクト読み込みで WAV (PCM) 音源以外読み込まれず無音になる
  - プロジェクト読み込みにおいてプラグインによるデコードが必要な音源の読み込みができていなかったことが原因
- Windows 版で WAV (PCM) 音源をファイルダイアログから開くことができない

## 12.18 30.0.0 (2020/8/12)

### 12.18.1 機能追加

- マウスの中央ボタンを押しながらのカメラ移動の追加
  - もともと MMD にあったが nanoem では未実装だったため実装
  - 右上のカメラ移動のアイコンの挙動と同じだが移動幅を抑えるように調整

### 12.18.2 仕様変更

- プロジェクト (nmm 形式) に保存するパスを絶対パスから相対パスに変更
  - 今回の変更により例えば DropBox のようなオンラインストレージでファイル同期を利用した際に別のマシンでも読み込めるようになります
  - 29.2.0 以前に保存したプロジェクトファイルは従来どおり読み込むことが可能ですが、新規プロジェクトでは相対パスで保存されます
    - \* 設定の切り替えは「[保存時のファイルパス設定](#)」で可能です
- プロジェクト読み込み時にアクセサリ、モデル、モデルの材質に割り当てられたエフェクトの中身がプロジェクト保存時とは違う内容になっていた場合エラーを出すように変更
  - 従来はスキップして読み込まれましたが気づけずファイル消失と勘違いする問題があるため明示的にエラーを出すようにしました
- カメラ変形方式の初期値を Global から Local に変更
  - MMD において Local が初期値であるため追従しました
  - モデルは従来どおり Local が初期値です
- 画像または動画の書き出しの際にプロジェクト保存を確認するかのダイアログを挟むようにした

- 動画出力中に落ちる問題が発生することによる巻き戻りが起こることを防ぐために追加
- 画像書き出しの際に未登録のボーンまたはモーフがある場合エラーを出すようにした
  - 画像書き出しは仕様上フレーム移動が発生するため未登録のボーンまたはモーフがあると消失する問題の対策として追加

### 12.18.3 不具合修正

- プロジェクト初期化で落ちることがある
- 編集及び再生中のフレームレートの「無制限」が機能しない
- IK 無効化したあと IK リンクの回転がゼロ初期化されずに残り続ける
  - 関連で IK 有効化無効化の切り替えを即座に反映されるようになりました
- IK リンクのボーンが IK ボーンの後ろにあると正常に変形しない
- 右下のハンドル経由でのカメラ移動において Local が Global と同じ扱いで処理されていた
- 右上アイコンによるカメラ移動が MMD のそれと違う挙動になっていた
- シフトキー押したときカメラの移動またはズームができない
  - 背景動画操作と干渉してたことが原因であったため、背景動画が読み込まれないときはカメラの高速移動または高速ズームを行えるように修正
  - 背景動画が読み込まれたときは挙動が上書きされるため注意が必要です
- モデルプラグイン実行後に保存されるモデルが MMD で読み込めないものになる
- シフトあるいはコマンド（コントロール）キーを利用したハンドルの移動または回転をすると位置が飛ぶ
- Windows 版において明示的に拡張子をつけないとファイル保存に失敗する
  - 拡張子がなくてもファイル名だけで保存できるように修正しました

## 第 13 章

# 既知の不具合

### 13.1 「トラブルシューティング」に記載

- プロジェクト保存に失敗する
  - [プロジェクトが保存出来ないのですが](#) を参照
- カメラモーションを読み込むと X 軸が反転する
  - [カメラモーションを読み込んだら角度の X 軸が反転する](#) を参照
- 動画出力するとフレーム欠けが起こる
  - [動画出力で書き出すと再生では起こらなかったフレーム欠けが起こる](#) を参照
- VRM から変換したモデルを読み込むとボーン表示になる
  - [VRM から変換したモデルを読み込ませるとボーン表示のみになる](#) を参照
- macOS 10.15 Catalina 以降で「キーボード入力監視」の確認を求めるダイアログが表示される
  - [macOS 10.15 以降で「キーボード入力監視」の確認を求めるダイアログが表示される](#) を参照
- macOS 10.12 Sierra で起動直後に落ちる
  - [nanoem が開かない、起動しない](#) を参照

### 13.2 対応予定なし

- Apple Software Renderer を利用した場合起動しない
  - OpenGL 系でかつ特定条件下でしか Apple Software Renderer が使われることがないため

## 13.3 現在進行系

- 補間曲線のコピーペーストが出来ない
  - 上記は未実装
- モデル読み込みで落ちる？
- モデル選択で落ちる？
  - クラッシュメトリクスで確認してるものの、発生条件がわからないため [不具合報告の方法](#) にある手順で報告をお願いします
- 動画出力における音ズレ？
  - 1.23.0 でおそらく修正
  - 少なくとも動画出力中に固まってしまうと固まった分も含まれてしまう問題はある
  - 情報不足のため要フィードバック
- PMD から読み込んだ時モデルの動きがおかしくなることがある

## 13.4 既知の不具合だったもの

### 13.4.1 25.0.0 で修正

- キーフレームをドラッグしても任意のフレームに移動出来ない

### 13.4.2 1.23.0.0 で修正

- SDEF 未実装

### 13.4.3 1.22.5.0 で修正

- macOS 10.11 以上かつ Metal が利用不可な場合起動直後に落ちる
- wav 形式以外の音源を読み込みして動画書き出しをすると落ちる
- アンチエイリアスを有効にして画像書き出しすると意図しないものになる



#### 13.4.4 1.22.3.0 で修正

- 音量調整ができない

#### 13.4.5 1.22.2.0 で修正

- レンダラが Metal の場合画像及び動画書き出しができない
- macOS が 10.13 以下で起動直後に落ちる
  - 関連で Metal が強制され、OpenGL が使われない

#### 13.4.6 1.22.1.0 で修正

- 起動直後に画面が表示されない
  - 1.22.0.0 で発生

#### 13.4.7 1.22.0.0 で修正

- macOS 10.14 Mojave 以降で実行すると 10.13 以前と比較して高負荷になる
  - nanoem 1.22 で対処予定
  - どうしても 1.21 以下で利用したい場合は Mojave へのアップグレードを控えてください
- Retina の高解像度モードではなく低解像度モードで動作してた

#### 13.4.8 1.21.0.0 で修正

- カテゴリ単位でモーフが存在しないモデルを操作するときモーフパネルからウェイトを変更したり登録ボタンを押すと落ちる
- ボーンが一切存在しないモデルを読み込むと落ちる

#### 13.4.9 1.20.0.0 で修正

- セルフシャドウが描画されない
  - おそらく 1.19.0 からの後退バグ

### 13.4.10 1.19.4.0 で修正

- モーションを読み込むと物理演算が無効にされる
- 特定のモデルでテクスチャが正しく描画されないことがある
  - 1.17 から発生
  - 確認されてるモデルとしてアリシア・ソリッドがあり、服装が正しく描画されない

### 13.4.11 1.19.3.0 で修正

- nanoem 形式で保存すると空になる
  - プロジェクト保存実装のバグ起因

### 13.4.12 1.19.2.0 で修正

- nanoem 形式 (nma 形式ではない) でのプロジェクトが保存できない
  - 読み込みからではなく新規で保存するとファイルが作成されない問題がある
  - 1.20 で修正予定
  - 1.19 以前は nma 形式での保存をお願いします
- ボーンモーフが機能しない
  - セルフシャドウが描画されない不具合と同じ後退バグ
- 「再生」したとき 60FPS で設定しても 30FPS 相当で再生される
  - 内部の時間計算の不具合が原因

### 13.4.13 1.19.0.0 で修正

- プロジェクト未保存時に出るダイアログの後に落ちる
- ボーン選択してコピーすると落ちる
- 物理演算設定で OK を押すと落ちる
- 動画出力時にキャンセルすると落ちる ( ? )
  - 原因と思われる箇所をいったん暫定対策したが防げてるかどうか事象が再現出来てないため ( ? ) をつけてる

#### 13.4.14 1.18.0.0 で修正

- 特定のモデル及びモーションを読み込むと「ユニコード文字列のデコードに失敗しました」と出る
- ボーンを動かした後元に戻ってしまう
  - 1.17.0.0 から 1.17.3.0 まで発生してた
  - 表示上の不具合でボーン登録することにより変形後の状態で登録される

#### 13.4.15 1.17.3.0 で修正

- モーション読み込みのところで反応がなく、ダイアログが開かないためモーションが読み込めない
  - 1.17.0.0 から 1.17.2.0 まで発生してた

#### 13.4.16 1.17.2.0 で修正

- Retina 環境で起動直後に落ちる
  - 1.17.0.0 から 1.17.1.0 まで発生してた

#### 13.4.17 1.17.0.0 で修正

- 起動直後から常時高負荷になる
  - 1.15.0.0 から 1.17.0.0 まで発生してた
- UI の文字表示が「？」に置換されることがある
  - 「～」(波チルダ)は変換の問題で「？」に置換されます
- 背景動画の描画が遅い

#### 13.4.18 1.15.0.0 で修正

- 外部親とアクセサリの情報がプロジェクトに保存されない
  - アクセサリ情報に関しては静止状態での位置情報は保存されますがモーションとして保存されません

#### 13.4.19 1.6.0.0 で修正

- 一度プロジェクト開いて閉じて同じプロジェクトを再度開こうとしても開けない

- その場合は一旦アプリケーションを終了させてからプロジェクトを開いてください

## 第 14 章

# macOS で MMD をつかう方法

---

重要: 単に macOS 上で nanoem を使うだけであればこのドキュメントを読む必要はありません

---

---

注釈: この部分だけ Mac の話ではなく Windows PC の話です

- MMD を快適に利用したいのであればラップトップ（ノート）ではなくデスクトップを推奨します
  - MMD を利用するのに重要な高性能な GPU が利用できるからです
  - どうしてもラップトップにする場合でも最低 15 インチ以上を選ぶべきです
    - \* これより小さいと物理的な制約から高性能な GPU が利用できません
- MMD を快適に利用する Windows マシン選びのポイントとして
  - 選び方がわからない場合はゲーミング用または VR Ready なものを選びましょう
  - 購入予算は最低でも 12 万円、できれば 20 万円以上を積んだほうがよいでしょう
    - \* 安くてももちろん動きますが MMD 用途だとエフェクトをたくさん利用したいという目的がある場合は後悔する可能性があります
  - 予算をかけられないかつラップトップしか選べない場合は「学生ゲーム開発者向け 低予算・高性能ノート PC」から探すとよいでしょう
- 詳しくは「MMD のパソコン選び」を参照してください

2021 年 1 月末に Microsoft からの DirectX9 のエンドユーザランタイムのインストーラのダウンロードが一時的に不可能になりました<sup>\*1</sup> が、措置を加えた形で現在はダウンロード可能になっています。

---

<sup>\*1</sup> <https://www.gamespark.jp/article/2021/02/19/106238.html>

x86 が生き続ける限りは MMD が動く環境がなくなることはないと思いますが、Mac の Apple Silicon 移行を考えると Windows も他 CPU の流れに沿う可能性は否定できず、今後数年間で「期待通りに」動く環境が減る可能性はあります。(MMD が取り巻いてる厳しい状況については「[よろずのおはなし](#)」を参照)

---

nanoem とは直接関係ないですが、macOS 上で MMD を使う方法を説明するページです。

## 14.1 BootCamp をつかう

---

注釈: [Intel Mac](#) でのみ利用可能な方法になります。Apple Silicon Mac では利用できません。

---

macOS に付属する BootCamp を使ってその上で Windows を動かして MMD を使う方法。Windows メインで macOS をたまたに動かす程度ならこの方法が最適でしょう。

- 長所
    - Mac でありながら Windows マシンとして使うことができる
    - 最大のパフォーマンスを活用できる
  - 短所
    - Windows のライセンス購入必須
    - macOS と Windows の切り替えが起動時しかできない
    - BootCamp 専用のパーティションを切る必要がある
    - macOS のバージョンと対応する Windows のバージョンが強く紐付けられる
    - [外付け GPU](#) に対応していない
- \* MacBook Air または MacBook Pro で利用する場合はこの点に注意が必要です

## 14.2 仮想マシンをつかう

---

注釈: BootCamp とは理由が異なりますがこちらも Intel Mac でのみ利用可能な方法になります。Apple Silicon Mac にも対応する可能性はありますが、仮に対応したとしても

- 2021 年 3 月現在で単体販売されておらずかつ Insider のみ提供の ARM 版 Windows を利用する必要がある

- x86 版の Windows を利用することが出来ない<sup>\*2</sup>
- x86 のエミュレーション動作になるため遅くなる
- 特に物理演算で性能影響が出やすい

以上から仮に対応したとしても素直に x86 CPU ベースの PC を使ったほうが安定動作を期待できるため推奨しません。

---

無償または有償の仮想マシンソフトウェアを利用してその中で Windows をインストールして MMD を使う方法。仮想マシンの代表的なソフトウェアとして以下があります（ðŸŒŽ 是有償版のみ）。

- VMware Fusion ðŸŒŽ
- Parallels Desktop ðŸŒŽ
- VirtualBox

利用実績から実質的に選択肢は上 2 つのいずれかになるでしょう。macOS をメインにつかいつつ Windows を利用するならこの方法がよいでしょう。

### 14.2.1 長所

- 切り替えが簡単で macOS を動かしながら Windows を使うことが出来る
- パーティションを切る必要がなく他の Mac への移行がファイルコピーで済む
  - ただしファイル移動させて他の Mac で動かす場合 Windows のライセンスの再認証が必要になります
- Windows のバージョンに依存しない

### 14.2.2 短所

- Windows のライセンス購入必須
- （有償の場合）仮想マシンのライセンス購入が必須
- 一回仮想マシンを経由する関係でパフォーマンスが BootCamp に比べると劣る

## 14.3 Unity + MMD4Mechanim をつかう

ゲームエンジンのひとつである Unity とそのプラグインである MMD4Mechanim 経由でモデル及びモーションをインポートする形で動かす方法です。

---

<sup>\*2</sup> <https://news.mynavi.jp/article/20210503-1882583/>

### 14.3.1 長所

- Windows を使わないためライセンスコストが一切かからない
- VRM 出力の環境を整備できる

### 14.3.2 短所

- インストールサイズが非常に大きい (2GB 以上)
- 操作方法が全く異なるため、覚え直さないといけない
  - Unity は最近のバージョンで日本語版も選べるようになったが基本的に UI が英語表記
    - \* 検索で引っかかるドキュメントも大抵は英語表記が前提です
  - MMD4Mechanim はあくまでモデルまたはモーションを読み込めるようにするまででそこからモーションを作成して出力することまではできません
- モデラーのポリシーによってこの方法が使えないことがある
  - モデル付属の README を確認して Unity 上で使えるかどうかを必ず確認してください
    - \* Unity 上の利用はたいてい目的外利用とみなされます
  - 上記の問題は主に女性向け二次創作モデルで引っかかることが多いです

## 14.4 Wine をつかう

Windows API をエミュレーションして Windows アプリケーションを動かすソフトウェアである [Wine](#) を使って動かす方法です。商用版として [CrossOver Mac](#) がありますが、実質的に英語版のみです。

### 14.4.1 長所

- Windows を使わないためライセンスコストが一切かからない
  - macOS 上でそのまま動かせる

### 14.4.2 短所

- アプリケーションのルックアンドフィールが macOS とは大きく異なる
- 利用実績が圧倒的に少ないためその分情報が少ない



- 動かなかったときの対処方法は自分で調べないといけない
- AppDB によると動作評価は Bronze または Silver



## 第 15 章

# Open Source Libraries

This software uses below libraries.

- [ALURE](#)
  - [MIT](#)
- [bimg](#)
  - [BSDL](#)
- [Bullet Physics](#)
  - [zlib](#)
- [bx](#)
  - [BSDL](#)
- [Font Awesome](#)
  - [SIL OFL 1.1](#)
- [ffmpeg](#)
  - [LGPLv2](#)
- [GLM](#)
  - [MIT](#)
- [GenEi Font](#)
  - [SIL OFL 1.1](#)
- [gslang](#)
  - [BSDL](#)

- HockeySDK
  - MIT
- ICU
  - ICUL
- ini.h
  - PD/MIT
- Dear ImGui
  - MIT
- jo\_gif
  - PD
- Lemon
  - PD
- klib
  - MIT
- lz4
  - BSDL
- minimp3
  - CC0
- minizip
  - zlib
- nuklear
  - PD
- OpenAL Soft
  - LGPL
- par
  - MIT
- parson

- MIT
- protobuf-c
  - BSDL
- sokol
  - zlib
- SPIRV-Cross
  - Apache License 2.0
- SPIRV-Tools
  - Apache License 2.0
- STB
  - PD/MIT
- zlib
  - zlib

## 15.1 Internal Libraries

- nanodxm
- nanoem
- timeline
- undo
- fx9

## 15.2 ffmpeg build configuration

```
mkdir -p build
cd build
../configure \
  --prefix=$PWD/install-root \
  --disable-debug \
  --disable-yasm \
  --disable-static \
```

(次のページに続く)

(前のページからの続き)

```

--disable-doc \
--disable-htmlpages \
--disable-manpages \
--disable-podpages \
--disable-txtpages \
--disable-bzlib \
--disable-iconv \
--disable-lzo \
--disable-network \
--disable-schannel \
--disable-sdl \
--disable-symver \
--disable-xlib \
--disable-zlib \
--disable-lzma \
--disable-securetransport \
--disable-avdevice \
--disable-avfilter \
--disable-postproc \
--disable-demuxers \
--disable-muxers \
--disable-decoders \
--disable-encoders \
--disable-bsfs \
--disable-parsers \
--disable-programs \
--disable-hwaccels \
--disable-filters \
--disable-devices \
--disable-protocols \
--enable-shared \
--enable-rpath \
--enable-small \
--enable-swresample \
--enable-swscale \
--enable-demuxer=avi,matroska \
--enable-muxer=avi,matroska \
--enable-decoder=flac,pcm_s16le,png,rawvideo,utvideo \
--enable-encoder=flac,pcm_s16le,png,rawvideo,utvideo \
--enable-protocol=file

```

## 15.3 ICU build configuration

```

FLAGS="-DUCONFIG_NO_BREAK_ITERATION -DUCONFIG_NO_COLLATION -DUCONFIG_NO_FORMATTING -
↳DUCONFIG_NO_TRANSLITERATION -DUCONFIG_NO_REGULAR_EXPRESSIONS -mmacosx-version-min=10.
↳9"

```

(次のページに続く)

(前のページからの続き)

```
CFLAGS="$FLAGS -std=c99" CXXFLAGS="$FLAGS -std=c++11" ./configure \  
--prefix=$PWD/../../install-root \  
--with-data-packaging=static \  
--enable-static \  
--disable-dyload \  
--disable-shared \  
--disable-extras \  
--disable-icuio \  
--disable-layout \  
--disable-layoutex \  
--disable-tests \  
--enable-tools=yes \  
--disable-samples
```





## 第 16 章

# nanoem の技術的説明

---

重要: 単に nanoem を使うだけであればこのドキュメントを読む必要は全くありません。ここは nanoem のソースコードを読もうとする技術者向けに書かれています。

---

警告: コード全体で少なくとも 10 万行以上あり、ゲームプログラムあるいは小規模なゲームエンジンを読み解く位の読解難易度があります。もちろん、MMD/MME についても理解する必要があり、加えて依存ライブラリがそれなりにある上にそれらを最低限把握する必要があります。

ここでは nanoem のソースコードがどうなっているかを説明するドキュメントです。

おおまかに nanoem はふたつのコンポーネントに分かれています

- nanoem
- emapp

加えて OS 毎に固有の機能を実装した UI 層と依存ライブラリを集約した `dependencies` がありますが、nanoem と emapp を中心に説明します。

nanoem のソースコードは [GitHub](#) 上で公開されています。

### 16.1 nanoem

C 言語で書かれた PMD/PMX/VMD のデータ操作に特化したライブラリです。厳密な意味での nanoem とはこちらのことを指します。また、追加拡張として以下を付属しています。

- ミュータブルなデータ操作
- 文字変換 (CoreFoundation/ICU/Win32/Qt)

- PMM データ操作
- NMD データ操作
- 物理演算
- PMD -> PMX 変換
- JSON

nanoem の API は CoreFoundation の API を参考に設計していて原則イミュータブルな API を基本として提供しています。また、全てのオブジェクト（構造体）は直接メンバーにアクセスすることのできない不透明なものとして扱います。これにより ABI 互換を確保しています。

オブジェクトは `Model` と `Motion` に大別しており、`Model` は PMD/PMX 両方とも読み込める仕様ですが、`emapp` 側の実装により PMD については追加拡張で提供されている変換機能を使って PMX に変換してから扱います。`Motion` は VMD のみですが、拡張関数により NMD も読み込むことが可能です。両方とも同じインターフェースを持ちます。

追加拡張として提供されているミュータブルなデータ操作を使うとイミュータブルなオブジェクトを直接書き込む形でデータの書き換えが可能です。こちらはデータ書き換えだけでなく書込み可能なバッファオブジェクトに対して書き込むことが可能です。ミュータブルなオブジェクトは必ずひとつのイミュータブルなオブジェクトに対してのみ利用する必要があります。

### 16.1.1 オブジェクト階層

- `Model`
  - `Vertex`
    - \* 頂点
  - `Material`
    - \* 材質またはマテリアル
  - `Bone`
    - \* ボーン
  - `Constraint`
    - \* IK
    - \* BulletPhysics の `btConstraint` とは違うので注意
    - \* この下に IK リンクに相当する `ConstraintJoint` がある
  - `Morph`

- \* モーフまたは表情

- ・ BoneMorph
- ・ ボーンモーフ
- ・ GroupMorph
- ・ グループモーフ
- ・ FlipMorph
- ・ フリップモーフ
- ・ ImpulseMorph
- ・ インパルスモーフ
- ・ MaterialMorph
- ・ 材質モーフ
- ・ UVMorph
- ・ UV モーフ
- ・ VertexMorph
- ・ 頂点 モーフ

- Label

- \* カテゴリ

- RigidBody

- \* 剛体

- \* BulletPhysics の btRigidBody に対応する

- Joint

- \* ジョイント

- \* BulletPhysics の btConstraint に対応する

- SoftBody

- \* ソフトボディ

- \* BulletPhysics の btSoftBody に対応する

- Motion

- AccessoryKeyframe
  - \* アクセサリのキーフレーム
  - \* NMD のみで VMD は対応しない
- BoneKeyframe
  - \* ボーンのキーフレーム
  - \* NMD ではさらに追加で物理演算の切替情報を持つ
- CameraKeyframe
  - \* カメラのキーフレーム
- LightKeyframe
  - \* 照明のキーフレーム
- ModelKeyframe
  - \* モデルのキーフレーム
  - \* NMD ではさらに追加で外部親などの情報を持つ
- MorphKeyframe
  - \* モーフのキーフレーム
- SelfShadowKeyframe
  - \* セルフシャドウのキーフレーム

### 16.1.2 NMD について

NMD は VMD の上位互換として VMD を拡張した protobuf ベースのバイナリデータです。

- ボーン及びモーフ名に対する制約がない
  - VMD の場合は 15bytes 以内におさめる必要がある
- VMD と比較して概ね 10% 以上の削減が可能
  - 名前を数値の ID として管理しているため

VMD のデータを NMD としてそのまま保存できます。NMD を VMD に保存することもできますが、NMD にしか保存できない拡張情報は失われます。

データ仕様は nanoem/proto/motion.proto で定義されています。

### 16.1.3 エラーについて

ステータスを示す列挙型を使います。エラーが発生する可能性のある関数は必ずその列挙型を関数の最後の引数にとります。

### 16.1.4 カスタムアロケータについて

nanoem は組み込みでカスタムアロケータに差し替える機能を提供しており、nanoem 内では必ずカスタムアロケータを経由してメモリ確保及び解放を行います。emapp ではそのカスタムアロケータを利用してメモリリークをチェックします。

### 16.1.5 文字列について

文字列は ShiftJIS/UTF-8/UTF-16 を扱う必要があるため、文字列オブジェクトとして独立した存在で扱います。追加機能として文字列変換を提供していますが、こちらは事実上必須になっています。

文字列のファクトリーオブジェクトを経由する形で専用の関数を使って文字列オブジェクトと文字列をやり取りします。

### 16.1.6 ユーザオブジェクトについて

各オブジェクトには任意のオブジェクトに紐付けることができるユーザオブジェクトがあります。通常ユーザオブジェクトによる任意のオブジェクトには関与しませんが、オブジェクトのデストラクタが呼ばれるとそのタイミングで任意のオブジェクトに対する破棄を行います。

emapp において `Model` のオブジェクトはこれを利用して拡張データをもたせています。その一方で `Motion` のオブジェクトは単純に使う理由がないことから利用していません。

### 16.1.7 nanoem の実装ポリシー

- スペースのみかつインデントは 4
  - `.editorconfig` で定義
- C89 ベース
  - 変数の定義は関数の先頭で行う
- オブジェクトに相当する構造体は全て `opaque` とする
  - 構造体のメンバーアクセスは必ず関数を通じて行う

- メンバーを直接公開することを禁止
- 名前付けは OpenCL をベースにしたカスタム
  - 構造体の名前は `lower_snake_case`
  - 関数名は `nanoem` を先頭につけて `UpperCamelCase`
  - 定数は `UPPER_SNAKE_CASE`

## 16.2 emapp

C++ で書かれたアプリケーションのコアとなるライブラリです。nanoem の大半の処理はここに集中しています。nanoem と emapp を土台に、プラットフォーム毎の UI は UI 層に分離させるように設計しています。

- emapp は歴史的経緯から C++ の例外、RTTI 及び C++11 の一部機能 (nullptr) を除いてつかっていません
  - [コマンド](#) を参照
  - 共有ポインタや自動ポインタも使っていないため、ポインタ管理は厳格に行う必要があります
- 依存ライブラリの関係から C++14 対応のコンパイラが必要です

### 16.2.1 ライフサイクル

emapp のライフサイクルは比較的ゲームあるいはゲームエンジンに近いものになっています。

- アプリケーションの初期化
  - 各種ライブラリの初期化
  - プロジェクトの作成
  - 前回クラッシュが発生した場合はリカバリ処理を走らせるかを確認
    - \* ユーザが受け付けた場合はリカバリ処理を実行
- アプリケーションの終了が呼ばれるまでフレーム処理
  - 描画処理
    - \* シャドウマップを描画
    - \* エフェクトのオフスクリーンレンダーターゲットを描画
    - \* ビューポートを描画
  - ScriptExternal のためのモデルあるいはアクセサリの描画

- ・ 背景動画を描画
- ・ グリッドを描画
- ・ プリプロセスのエフェクトを描画
- ・ すべてのモデルのエッジを描画
- ・ すべてのモデル及びアクセサリの描画
- ・ すべてのモデル及びアクセサリの地面影を描画
- ・ ポストプロセスのエフェクトを描画
- ・ screen.bmp 専用のレンダーターゲットを転写
- \* 描画コマンドの一括処理
  - ・ 31.0 から導入
  - ・ 詳細は「描画コマンドの一括処理」の項目にて
- \* UI のアイコンなどを描画
- \* UI (ImGui) を描画
- \* ウィンドウに描画結果を表示
- \* リセット処理が要求された場合はリセット処理を実行
  - ・ 各種エフェクトのすべてのレンダーターゲットを再生成及び再設定
  - ・ ビューポートのレンダーターゲットを再生成及び再設定
- \* 2 フレームに 1 回 UI スレッドにダミーのイベントを通知
- \* 動画エンコード処理
  - ・ 動画エンコード処理が実行中の場合のみ
- \* プロジェクトの更新処理
  - ・ 音源の位置を更新
  - ・ 音源再生中の場合のみ
  - ・ シーク処理
  - ・ 物理演算前のモーシヨンの適用処理
  - ・ すべてのモデルに対して以下の順番で実行
  - ・ モデル（表示や IK 有効無効の方）のキーフレームを適用

- ・ モデルの材質のリセット
- ・ モデルのボーン変形をリセット
- ・ モデルのモーフをリセット
- ・ モーフのキーフレームを適用
- ・ 物理演算適用前のボーンのキーフレームを適用
- ・ ボーンのキーフレーム単位の物理演算の有効無効の切り替え処理
- ・ 物理演算に適用するためのボーンのパラメータを設定
- ・ 物理演算の実行
- ・ 物理演算後のモーションの適用処理
- ・ すべてのモデルに対して物理演算適用後のボーンのキーフレームを適用
- ・ すべてのアクセサリに対してキーフレームを適用
- ・ カメラのキーフレームを適用
- ・ 照明のキーフレームを適用
- ・ セルフシャドウのキーフレームを適用
- ・ カメラの更新
- ・ 照明の更新
- ・ モデルの変形処理
- 各種イベント処理
  - \* イベント処理中にエラーが発生したらエラーダイアログを表示
- イベント処理中にリセット処理が要求された場合は再度リセット処理を実行
- アプリケーションの終了
  - プロジェクトの破棄
  - 各種ライブラリの終了処理

## 16.2.2 主要要素



## アプリケーション

プロジェクト、描画、UI (ImGui)、入力のやり取りを一括管理するオブジェクト。  
`emapp::BaseApplicationClient` を通じて処理する。

スレッドに対応して UI 層とのやり取りの分離をはかる `emapp::ThreadedApplicationService` があり、  
Windows 版及び macOS 版ではこちらを利用する。 `emapp::ThreadedApplicationClient` を通じて処理する。

`emapp::BaseApplicationService` に対応する

## プロジェクト

すべてのモデル、モーション、アクセサリ、エフェクトを包括管理するオブジェクト。オフスクリーンを含めた全てのレンダーターゲットの描画及び破棄もここで行っている。

`emapp::Project` に対応する

## モデル

PMD/PMX に対応する描画対象オブジェクト。

`emapp::Model` に対応する

## アクセサリ

X に対応する描画対象オブジェクト。

`emapp::Accessory` に対応する

## モーション

動きを定義するオブジェクト。以下の種類があり、この内ひとつのみに所属する。

- モデル
- アクセサリ
- カメラ
- 照明
- セルフシャドウ

`emapp::Motion` に対応する

## カメラ

`emapp::ICamera` が対応する

## 照明

唯一の大域光源。MMD の仕様にあわせてディレクショナルライトのみ。

`emapp::ILight` が対応する

## エフェクト

MME 互換の複数のテクニック及びパスから構成されるオブジェクト。MME の技術仕様は MME に同梱している `REFERENCE.txt` を参照。

エフェクトの仕様が複雑でかつ他のオブジェクトにもかなり食い込んでるため、読解難易度は最も高いとみています。

`emapp::IEffect` が対応する

## テクニック

描画するための条件定義。複数のパスから構成される。

`emapp::effect::Technique` が対応する

## パス

頂点シェーダとピクセルシェーダをセットにした描画単位。

`emapp::effect::Pass` が対応する

## コマンド

巻き戻しが可能な操作単位。`undo.c` のユーザデータとして持っており、`undo.c` のコールバックを通じて巻き戻しあるいはやり直しが実行される。また、`undo.c` の永続化の仕組みを利用してアプリケーションがクラッシュしたときに直前の操作まで巻き戻す仕組みもコマンドが持っている。

キーフレーム登録あるいは削除、ボーンの移動やモーフの変更などの主要な操作はコマンドを通じて実行される。

複数のコマンドをひとつにまとめることが可能なバッチコマンドもあり、大規模なキーフレーム変更が発生するコマンドで利用している。

### 16.2.3 描画コマンドの一括処理

30.0 までは材質またはレンダーターゲット単位にレンダースを発行する処理になっていましたが Apple Silicon 対応において描画が崩れる問題があったため、レンダーターゲット単位にレンダースをまとめるように描画処理の見直しを実施しました。これにより無駄なレンダースを作らせないようにしたためパフォーマンス改善を可能になりました。

- `Project::SerialDrawQueue`
  - 原則として毎回レンダースを発行する
  - 例外として前회가 `SerialDrawQueue` でかつ同じレンダースの場合マージ可能ならマージする
  - 主にポストエフェクトで利用
- `Project::BatchDrawQueue`
  - レンダース単位にまとめる
  - 従来の描画は基本的にこちらを利用

描画コマンドの一括処理オブジェクトの管理は `Project` にあるものの、インターフェースである `emapp::sg::PassBlock::IDrawQueue` を経由するため中身は直接公開していません。

### 16.2.4 音源の同期補正処理

音源の位置とクロックオフセットを比較し、レイテンシの小さいほうを優先する同期補正処理が Windows 版と macOS 版に実装されています。具体的な流れは以下の通りです。

- 音源の位置をサンプルオフセットとして計算して取得
- クロックを秒に変換して音源の周波数と乗算しサンプルオフセットとして比較
- 上記二つの差分をレイテンシとして取得し、しきい値 (音源の周波数を 60FPS 基底に Windows 版では 15FPS macOS 版では 10FPS 相当で計算) で比較
  - レイテンシがしきい値よりも小さい場合はクロックを採用
  - レイテンシがしきい値よりも大きい場合は音源の位置を採用

通常はクロックを採用するものの、フレーム落ちにより処理が追いつかなくなった場合は強制的に音源の位置を採用する仕組みとなっています。

### 16.2.5 emapp の実装ポリシー

- スペースのみ、インデントは 4

- `.editorconfig` で定義
- C++ の例外は使用禁止
- C++ の実行時型情報 (RTTI) は使用禁止
- C++11 の `nullptr` 以外は使用禁止
  - ただし UI 層は例外的に C++11 の使用が認められる
  - UI 層でも C++14 以降の機能利用は認められない
- STL は原則として利用しない
  - かわりに同梱の `TinySTL` を利用する
  - ただし UI 層では一部利用 (`std::atomic`) している
- 名前付けは Qt/WebKit をベースにしたカスタム
  - クラス名は `UpperCamelCase`
  - メソッド名は `lowerCamelCase`
  - 定数は `k` を頭につけて `UpperCamelCase`
  - メンバー変数は原則として `m_` 接頭詞がつく
  - `protobuf` のような自動生成によるものは適用対象外

### 16.2.6 単体テスト

`Catch2` を利用した単体テストが `nanoem/test` 及び `emapp/test` にあります。リリース前は必ず全てのテストをパスする必要があります。

### 16.2.7 fx9

エフェクトをコンパイルするために作られた内製ライブラリです。fx9 はエフェクトプラグインを通じて利用されます。

- 出力するシェーダ言語を設定する。以下から設定可能
  - GLSL
  - MSL
  - HLSL
  - SPIR-V

- エフェクトのソースを入力
- AST に変換して SPIR-V 形式にコンパイル
  - fx9 がやることは文法をパースして AST に変換すること
  - 字句解析及び AST は `glslang` が提供するものを利用する
    - \* fx9 が実装しているのは DirectX のエフェクトの文法を解析処理と AST への変換処理である
    - \* 文法解析のバックエンドは `Lemon` を利用している
- 最適化が有効の場合は `SPIRV-Tools` でシェーダを最適化する
- 出力するシェーダ言語に応じて `SPIRV-Cross` で変換
- fx9 独自の protobuf 形式のバイナリデータで出力
  - データ仕様は `emapp/resources/protobuf/effect.proto` で定義
  - protobuf が `emapp` 上にあるため、独立したライブラリとしてまだ完全に分離できてない状態

### 16.2.8 nanodxm

DirectX の .x 形式のテキストデータをパースするために作られた内製ライブラリです。バイナリは未対応です。

独立したライブラリとして一応使うことが可能です。

### 16.2.9 undo.c

`emapp` で使われている `undo/redo` の操作に特化した内製ライブラリです。クラッシュ後の起動時に行われるリカバリ処理を実現するためにコールバックを通じた永続化にも対応しています。

独立したライブラリとして一応使うことが可能です。

### 16.2.10 sokol

<https://github.com/floooh/sokol> (実際にはフォーク版 <https://github.com/hkrn/sokol> を利用)

`nanoem` の描画バックエンドとして利用しているライブラリです。

基本的にはオリジナルの実装をそのまま利用しますが、デバッグやバッファの読み取りのために内部構造に直接アクセスして拡張しています。また、複数のバックエンドを切り替えられるようにするため、共有ライブラリとして組み込んでいます。

最初期は `bgfx` を利用していましたが、以下の理由から切り替えを行っています。ただし、関連ライブラリである `bx` および `bimg` は引き続き利用しています。

- バッファの明示的な上限設定が必要
- 内部構造上デバッグが困難
- バックエンドのオブジェクトへの直接アクセスができなかった

### 16.2.11 ImGui

<https://github.com/ocornut/imgui>

nanoem の GUI バックエンドとして利用しているライブラリです。

主にゲーム開発における GUI ライブラリとして利用されますが、nanoem では直接ユーザが利用する GUI ライブラリとして利用しています。初期から利用しておりその時はデフォルトのルックフィールを利用していましたが、`nuklear` を一時期に採用してた関係から `nuklear` の見た目と合わせる形でルックフィールを変更しています。

## 16.3 よろずのおはなし

### 16.3.1 MMD は DirectX11 に移行できるのか？

まずこの質問の背景として 2021 年 1 月末にセキュリティ上の理由で SHA1 署名のエンドユーザ向け DirectX9 のインストーラ及び「古い」DirectX9 SDK が Microsoft 公式ページから削除されました（[英語での解説記事](#)。もともと計画されてたもので時勢により延期されたため復活はないと見られましたが、少なくとも [英語版](#) は SHA2 に署名し直して再開したようです）。

一時期上記の混乱が発生したため MMD の公式ページから再配布可能なエンドユーザ向け DirectX9 インストーラを直接ダウンロードできる仕組みを取って対応しました。そのことからおそらくこの疑問が出るだろうと見て記しておきます。

---

注釈：意外に思われるかもしれませんが DirectX9 ランタイムそのものは Windows10 に含まれており、保守も続けられています。

一方で DirectX9 を利用する際のデファクトスタンダードであり MMD も利用しているライブラリである D3DX が Windows 10 に入っていないことから DirectX9 インストーラによる導入が必要となります<sup>\*1\*2</sup>。

---

<sup>\*1</sup> <https://walbourn.github.io/where-is-the-directx-sdk-2013-edition/>

<sup>\*2</sup> [https://support.steampowered.com/kb\\_article.php?ref=9974-PAXN-6252](https://support.steampowered.com/kb_article.php?ref=9974-PAXN-6252)

D3DX は歴史的経緯によりバージョン違いが多数あり適切に導入するのが非常に難しいことからインストーラ経由による導入が必須であり、D3DX 単体の再配布禁止の根拠となっています<sup>\*2</sup>。

以下の理由から少なくとも本体自身からはおそらく不可能と見ています。また、仮に運良く移行できたとしても DirectX9 のそれとは別物になる可能性が高いため、利用者が使ってくれるかどうかが未知数です。

- DirectX9 と DirectX11 とでは設計レベルで異なるため DirectX9 から DirectX11 に移行するには設計変更が必要
  - さらに MMD では D3DX を利用しているが DirectX11 では直接同等の機能を提供しておらず<sup>\*3</sup>、独自のものに置き換える必要がある
- MME で利用しているエフェクト文法が DirectX11 では互換性がないため利用できない
  - DirectX11 に対応したエフェクトを改めて書き起こす必要がある
  - DirectX のエフェクト形式そのものが非推奨のため今後利用できるかどうかが不透明
- そもそも MMD/MME 自体の開発が事実上停滞している状況にある

ただし MMD 本体からではなく MME の仕組みを利用して DirectX11 に対応させる取り組みはいくつかあります。

- MME と同じ仕組みで DirectX11 に対応させる MME の開発者自身による取り組み
  - 動画は [sm21860058](#)
  - あくまで DirectX11 の新機能を試すための試験的な取り組みからか、2021 年 2 月時点でそれ以上の動きはないようです
- 別の開発者による MME と同じ仕組みで DirectX11 に対応させる取り組み
  - 動画は [sm35941062](#)
  - 同開発者による MMDPlugin の仕組みを利用する形
  - エフェクトにも対応しているように見えるが HLSL Shader Model 5.0 ベースに MME の構文を適用できるようにした独自形式の様子

### 16.3.2 MMD は x86 以外に移行できるのか？

先に書いたとおり MMD は D3DX に依存しており、D3DX がクローズドソースであるため D3DX 利用を脱却するか D3DX が x86 以外にも対応しない限りは不可能とみられます。DirectX9 版の D3DX はすでに保守対象から外れており移行推奨の記事<sup>\*3</sup>があるくらいなので x86 以外の対応は絶望的と考えられます。

<sup>\*3</sup> <https://walbourn.github.io/living-without-d3dx/>

少なくとも ARM 版は x86 エミュレーションで対応できるとみられますが、性質上速度低下は避けられず特に物理演算がボトルネックになる可能性があると考えられます。