

スイート千鳥エンジン Ver.2

Operation Guidance 操作ガイダンス





本書のご使用にあたって

本操作ガイダンスは、おもに初めて本製品を利用する方を対象に操作の流れに沿って、操作、入力、処理方法を説明したものです。

ご利用にあたって

ご使用製品のバージョンは、製品「ヘルプ」のバージョン情報よりご確認下さい。 本書は、表紙に掲載のバージョンにより、ご説明しています。 最新バージョンでない場合もございます。ご了承下さい。

本製品及び本書のご使用による貴社の金銭上の損害及び逸失利益または、第三者からのいかなる請求についても、弊社は、その責任を一切負いませんので、あらかじめご承知置き下さい。 製品のご使用については、「使用権許諾契約書」が設けられています。

※掲載されている各社名、各社製品名は一般に各社の登録商標または商標です。

©2022 FORUM8 Co., Ltd. All rights reserved..

目次

第1章 製品概要 5 5 1 スイート千鳥エンジンの概要 6 2 フローチャート 7 第2章 操作ガイダンス 7 1 事前準備(Visual Stusdio 2019インストール) 2 チュートリアル 7 2-1 プロジェクトの読み込み 7 2-2 作業ディレクトリの設定 8

- 9 2-3 ライセンス認証
- 2-4 ビルド
- 10
- 2-5 実行 10 3 環境設定 11
- 11 3-1 プロジェクトの作成
- 3-2 プロジェクトの設定(構成:Debug、プラットフォーム:x64) 14
- 3-3 プロジェクトの設定(構成:Release、プラットフォーム:x64) 17
- 20 3-4 プロジェクトのビルド

第3章 操作ガイダンス(スイート千鳥エンジンエディター) 23

23 1 事前準備

- 1-1 ライセンス認証 23
- 1-2 アプリサイズの設定 23
- 2 プロジェクト作成 24
- 3 モデルの追加 24
- 4 モーションの設定 25
- 5 アプリ作成 26

29 第4章 Q&A

- 29 1 スイート千鳥エンジンについて
- 2 導入について 29
- 3 ライセンス認証について 30
- 4 スイート千鳥エンジンの機能について 31
- 5 スイート千鳥エンジンが対応しているファイル形式について 31
- 6 その他 31

1 スイート千鳥エンジンの概要

スイート千鳥エンジンは、Visual Studioでの開発で使用できるライブラリ集となっております。3Dモデルの表示から各種イベントの制御までを、タスクシステムにより容易に管理できます。また、最新のVisual Studioへの対応や、FBXモデルの取り込みに対応しています。これにより、学生などは無償で利用できるVisual Studio Communityでの開発ができたり、統合型3DCGソフトShade3Dで作成した3Dモデルを取り込んだりすることができます。

必要システム

本製品は、Windows 8.1/10環境を有するOS上で動作します。 また、本製品を使用して開発するためには、Visual Studio 2019が必要となります。 papetプラグインを使用してパーティクルエフェクトデータを作成するためには、Autodesk Maya 2020が必要となります。 なお、Autodesk Maya LT 2020ではpapetプラグインはご利用いただけませんのでご注意ください。

ライセンスについて

スイート千鳥エンジンでは、ライセンス認証を行った場合のみ、作成されたゲームが正しく表示されます。そのため、ライセンス認証を行うためのコードを記述いただく必要があります。

2 フローチャート



第2章 操作ガイダンス

1 事前準備(Visual Stusdio 2019インストール)

Microsoft Visual Studio Yearl 9x36 2019 BBE エディンタン ダウンロード サポート マサブスクライバーアクセス Visual Studio MBBB Tへての Monorh B

Visual Studio Community Web アプリケーションとクラウト サービスだけでなく、Windows、Android、ios 向けのモダン アプリケー ションを作成するためのフル機能を装備した、鉱気可能な無料の IDE です。 Wind State のダフジロート: Visual Studio Community 2019をインストールします。 以下URL内の「Visual Studioのダウンロード」 ボタンよ りダウンロードできます。

https://docs.microsoft.com/ja-jp/visualstudio/ releases/2019/release-notes

※Visual Studioは、製品版(Professional, Enterprise)の 2019をお持ちの場合は、そちらをご使用いただいても 構いません。

なお、2019以外のバージョンを使用すると、正しく動作しない場合があるため、必ず2019を使用してください。

※Visual Studio 2019のインストール時には、「C++ に よるデスクトップ開発」にチェックを入れてください。

2 チュートリアル



チュートリアル「Tutorial.sln」を例題として作成します。 本ドキュメントでは、モデルの表示とモーションの再生 を行うチュートリアルプロジェクトの使用方法と、コード 内容を説明します。

■各入力項目の詳細については製品の【ヘルプ】 をご覧 ください。

2-1 プロジェクトの読み込み

ローカル ディスク (C:) > SuiteChidori	ingine > tutorial > Windows		
名前	更新日時	種類	サイズ
.vs	2021/08/25 13:28	ファイル フォルダー	
data	2021/08/25 11:21	ファイル フォルダー	
include	2021/08/25 11:21	ファイル フォルダー	
src	2021/08/25 11:21	ファイル フォルダー	
🔄 Tutorial	2021/08/25 13:33	ファイル フォルダー	
x64	2021/08/25 13:33	ファイル フォルダー	
🖓 Tutorial.sln	2020/08/17 15:00	Microsoft Visual S	2 KB

SuiteChidoriEngineのtutorialフォルダ内のWindowsフォルダを開き、「Tutorial.sln」をダブルクリックしてTutorialプロジェクトを開きます。

2-2 作業ディレクトリの設定



Visual Studio2019が開かれるので、右側「ソリューションエク スプローラー」-「Tutorial」を右クリックし、「プロパティ」をク リックしてプロパティページを開きます。

成(<u>C</u>): アクティブ(Debug)	✓ ブラットフォーム(P):	x64 ~	構成マネージャー(
 3 環境プロ(ティ 金融 デドップ) シーンディングトリー レーンディンクトリー レーンプレーシー レーンプレーシー レーンプレーシーシーシーシーシーシーシーシーシーシーシーシーシーシーシーシーシーシーシ		S(SolutionDiv)S(Platform)¥S(Configuration) S(Platform)¥S(Configuration)¥ S(ProjectWanne) アブリブ-730 (cere) 10.0 (機能のゲンス)-N=k1ている/(-ジョン) Visual Studio 2019 (vf42) 税定 (SC C + 14 標準) 税定 (法承の MSVC)	1)4
	出力ディレクトリ コンパイル済みプログラムが配置される場	所のバス。環境変数を含めることができます。	



構成(C):で、「Debug」「Release」を切り替える事が出来ま す。 「プラットフォーム」を変える必要はありません。

<mark>構成(C):</mark> アクティブ(Debug)

プラットフォーム x64

「構成プロパティ」-「デバッグ」-「作業ディレクトリ」 下矢印選択→<参照...>をクリック→下記フォルダを選択 C:¥SuiteChidoriEngine¥tutorial¥Windows¥data

式(C):	Release	~ ブラットフォ	:-/д(Р): x64	~	構成マネージャー	·(0).
構成	เวืองกระ	起動するデバッガー:				
3	全般	ローカル Windows デバッガー				
	手相					
	アハッソ	אעקב	\$(TargetPath)			
ъ	C++	コマンド引数				
Þ	リンカー	作業ディレクトリ	¥data			
•	マニフェスト ツール	アタッチ	<編集>			
Þ)	(ML ドキュメント ジェネレーター	デバッガーの種類	<参照>			
Þ	ブラウザー情報	環境	<親またはプロジ	ェクトの既定値から継承>		
Þ	ビルドイベント	マージ環境	1919			-
ÞĴ	カスタム ビルド ステップ	SQL デバック	いいえ			
Þ	3ド分析	AMP の既定のアクセラレータ	7 WARP ソフトウ:	ェアのアクセラレータ		
		作業デルクトリ				

構成(C):

Release

プラットフォーム x64

「構成プロパティ」-「デバッグ」-「作業ディレクトリ」 下矢印選択→<参照...>をクリック→下記フォルダを選択 C:¥SuiteChidoriEngine¥tutorial¥Windows¥data

設定が完了したら「OK」をクリックしてプロパティ画面を閉じます。

構成(C): Debug ✓ プラットフォーム(P): x64 ~ 構成マネージャ−(O)... 起動するデバッガー: ▲ 構成プロパティ 構成プロ(ティ 金校 詳細 デバッグ
 VC++ディレクドリ
 VC++ディレクドリ
 VC++ > リンカ
 マニフェストッール
 マニフェスト・ジェネレー・
 ブラングー指載
 ブラングー指載
 レスタムとレンステップ
 コード分析 ローカル Windows デバッガー コマンド コマンド引数 作業ディレクトリ アタッチ \$(TargetPath) \$(ProjectDir)...¥data いいえ デバッガ 環境 マージ現 SQL デ AMP の ? プロパティ ページへの変更を保存しますか? アクセラレータ はい(Y) いいえ(N) キャンセル 作業ティレクトリ アブリケーションの作業ディレクトリです。既定でディレクトリにはプロジェクト ファイルを含みます。 OK キャンセル 適用(A)

2-3 ライセンス認証



アクティベーションファイル(Chidori.ls)を作業ディレクトリ に設置し、右側「ソリューションエクスプローラー」-「main」-「main.cpp」をクリックすると、作業ディレクトリに表示されま す。

main.cppのwinMain関数内のactivateLicense(" ")の引数に シリアルコードを記述します。

シリアルコードはハイフンも含めて記述してください。

※アクティベーションファイル(Chidori.ls)とシリアルコードの素 用方法について (Q3-1.参照) https://www.forum8.co.jp/faq/win/chidori-qa.htm#q3-1

※「プロパティページへの変更を保存しますか?」 とメッセージ が出たら「はい」 をクリックして下さい。

2-4 ビルド



□ーカルディスク(C) > SuiteChidoriEngine > tutorial > Windows > x64 > Debug
 名前 ^ 更新日時 種類 サイズ
 ③ libfbxsdk.dll 2021/07/11:50 アブリケーション拡張 8,288.K8
 Ⅲ TutoriaLexe 2021/08/25 13:33 アブリケーション 2,461.K8
 ☑ TutoriaLpdb 2021/08/25 13:33 Program Debug D... 4,004.K8

新しいソリューション エクスプローラーのビュー(N)

プラットフォームを「x64」に変更して、右側「ソリューションエ クスプローラー」-「Tutorial」を右クリックし、「ビルド(U)」をク リックします。

libfbxsdk.dllの設置

生成された.exeファイルと同じディレクトリに「libfbxsdk.dll」を 設置します。 ※「libfbxsdk.dll」は「(インストール先)¥lib¥dll¥x64」にあるも のを使用します。

2-5 実行



「Visual Studio 2019」の画面に戻って「F5」を押して実行して ください。 成功するとアプリが起動し、動くモデルが表示されます。

3 環境設定

スイート千鳥エンジンでアプリを作成するための環境設定を行います。

3-1 プロジェクトの作成

近開いた項目(R)		開始す	3
<u>م</u>		*	リポジトリのクローン(C) GitHub や Azure DevOps などのオンライン リポジトリナ 6コードを取得します
Tutorial.sin C#SuiteChidoriEngine#tutorial#Windows	2021/08/25 14:06	Ċ	プロジェクトやソリューションを開く(P) ローカルの Visual Studio プロジェクト表たは .sln ファイル を戻きます
GameProject.sin C#Users#forum@Ksource%repos#GameProject	2021/08/25 11:42	2	ローカル フォルダーを開く(F) 任意のフォルダー内のユードに移動して編集します
CitUsersWorum8KsourceVreposWProject1	LOL IVOP LA TILLA	*3	新しいプロジェクトの作成(N) 開始するには、コードスキャフォールディング付きのプロジェ クトテンプレートを選択します
			□-ドなしで統行(W) →

「Visual Studio 2019」を起動し、「新しいプロジェクトの作成 (N)」をクリックします。



「空のプロジェクト」を選択して、「次へ(N)」 をクリックしま す。

新しいプロジェクトを構成します		
空のプロジェクト C++ Windows コンツール		
プロジェクト名(J)		
GameProject		
場所(1)		
CHSuiteChidoriEngine	w	
ンリューション名(M) ()		
✓ ソリューションとプロジェクトを同じディレクトリに配置する(D)		

 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10
 10<

「プロジェクト名(N)」「場所(L)」(任意)を設定し、「ソリューションとプロジェクトを同じディレクトリに配置する(D)」を チェックして「作成(C)」をクリックします。

プロジェクト名

GameProject

場所 C:¥SuiteChidoriEngine¥

ソリューションとプロジェクトを同じディレクトリに配置する(D) チェックON

プロジェクトが開いたら、右側「ソリューションエクスプロー ラー」に表示されている下記フィルターを削除します。

「ソース ファイル」 「ヘッダー ファイル」 「リソース ファイル」

第2章 操作ガイダンス

> ローカル ディスク (C:) > SuiteChidoriEngine > GameProject

名前	更新日時	種類	サイズ
.vs	2021/08/25 17:02	ファイル フォルダー	
📙 include	2021/08/25 17:10	ファイル フォルダー	
src src	2021/08/25 17:10	ファイル フォルダー	
📲 GameProject.sln	2021/08/25 17:02	Microsoft Visual S	2 KB
💁 GameProject.vcxproj	2021/08/25 17:02	VC++ Project	8 KB
GameProject.vcxproj.filters	2021/08/25 17:02	VC++ Project Filte	1 KB
GameProject.vcxproj.user	2021/08/25 17:02	Per-User Project O	1 KB



インストール済み 並べ替え 既定 II' 🗉 種類: Visual C++ C++ ヘッダー ファイルを作成します ▲ Visual C++ sual C++ ヨード 号式設定 データ リリース Web ユーティリティ プロパティシー HLSL グラフィックス C++ 77/JU (.cpp) Visual C++ [h] ヘッダー ファイル (.h) Visual C++ C++ 75X Visual C+ C++ モジュール インターフェイス ユニット (.ixx) Visual C++ ▶ オンライン 名前(N): 場所(L): 参照(<u>B</u>)... C:¥Suite (GameProject¥include) 遠加(A) キャンセル

g >+Lo mao ave ou 0 - 0 0 - 6 m ≥ 1 > 7 1	Robe Die Hole nie see see see see sol was voor die sooo	2 Geodese			+000 Å 1010020-000 Å 0 0 4 3 5 12 2 4 9 0 0 1010020 + 088 0+ 0 0 10100 € 0810 4 0 0 10100 € 0810 4 0 0 10100 € 0810 € 0 0 10100 € 0810 € 0 0 10100 € 0810 € 0 0 10100 € 0810 € 0 0 10100 € 0810 € 0 0 10100 € 0810 € 0 0 10100 € 0810 € 0 0 10100 € 0810 € 0 0 10100 € 0810 € 0 0 10100 € 0810 € 0 0 10100 € 0810 € 0 0 10100 € 0 0 10100 € 0 0 0 0 0 0 0 0	- 0) # **
			D Multiple. D BROUGHOM. D BROUGHOM.	Det-Bullt-A Dett-Alt-A		04-548-X 04-548-X 04-X 04-7 04 82 82 82 82
13					Y712-010日第344759 #第5-95-333-488 #第5-95-333-4597 # 25320-3-25865-488(0) D -5-25858(# 260540) # 260540	A3-54w
anto-					0640 705127-659321.87.	

エクスプローラーを開き、プロジェクトフォルダに 「include」 「src」 という名前のフォルダを新規作成します。

「Visual Stusdio 2019」を画面に戻り、右側「ソリューションエ クスプローラー」-プロジェクト名(GameProject)を選択し、右ク リックで「追加(D)」-「新しい項目(W)」をクリックします。

「新しい項目の追加」画面が開くので、設定を行います。

<mark>ファイル</mark> ヘッダー ファイル (.h)

.....

<mark>名前</mark> stdafx.h

場所

C:¥SuiteChidoriEngine¥GameProject¥include¥ (先ほどプロジェクト下に作った「include」フォルダを選択しま す)

「追加(A)」をクリックします。

右側「ソリューションエクスプローラー」-プロジェクト名 (GameProject)を選択し、右クリックで「追加(D)」-「新しい項 目(W)」をクリックします。









「新しい項目の追加」画面が開くので、設定を行います。

<mark>ファイル</mark> C++ ファイル (.cpp) 名前 stdafx.cpp

場所

C:¥SuiteChidoriEngine¥GameProject¥src¥ (先ほどプロジェクト下に作った「src」フォルダを選択します)

「追加(A)」 をクリックします。

※今後は「ヘッダー ファイル」「C++ ファイル」はそれぞれ 「include」「src」に追加していきます。

右側「ソリューションエクスプローラー」-「stdafx.h」をクリックし、作業ディレクトリに下記コードを入力します。

/**

stdafx.h */ #pragma once #ifndef_PA_STDAFX_H_ #define_PA_STDAFX_H_ #include <chidori/chidori.h> using namespace chidori; #endif // !_PA_STDAFX_H_

右側「ソリューションエクスプローラー」-「stdafx.cpp」をクリックし、作業ディレクトリに下記コードを入力します。

/** stda

stdafx.cpp */

#include "stdafx.h"

右側「ソリューションエクスプローラー」-「stdafx.cpp」をク リックし、右クリックで「プロパティ(R)」を選択してプロパティ ページを開きます。







プラットフォーム x64

「C/C++」-「プリコンパイル済みヘッダー」

プリコンパイル済みヘッダー 作成 (/Yc)

「構成(C):」を「Release」に変えて、同じく下記設定を行いま す。

「C/C++」-「プリコンパイル済みヘッダー」

プリコンパイル済みヘッダー 作成 (/Yc)

設定が完了したら「構成(C):」を「Debug」に戻し、「OK」ボタンをクリックして画面を閉じます。

3-2 プロジェクトの設定(構成:Debug、プラットフォーム:x64)



に J : Debug、 J フ ット J オ ーム: x64)「構成:Debug」「プラットフォーム:x64」になっている事を確

「痛成:Debug」「フラットフォーム:X64」になっている事を確 認し、右側「ソリューションエクスプローラー」-プロジェクト名 (GameProject)を選択し、右クリックで「プロパティ(R)」をク リックします。



GameAngent プロパタイページ パレパタイ パレパタイ 単純 単純 プリフトフォーム(P): %64 単二、単成マネージャー(O)… ・ 塩点 単純 パレック・ロシー ※64 ● 塩点マネージャー(O)… ・ 塩点 日油 パレック・ロシー ※52 一日山 シュルタ 高年をを用する 1/100 ● ・ ビバー・ディレクトリ ・ ビビ・ディレクトリ ・ ビビ・ディレクトリ ・ ビビ・ディレクトリ ・ ビビ・ディレクトリ ・ ビブ・ジャー・ ・ ブリコレ ロック・ロ・ フレーム ポイシック・ロ・ フレーム ポイシック・ロ・ フレーム ポイシック・ロ・ フレーム ポイショー・ ・ ブリコレ ロック・ロ・ フレーム ポイショー・ ・ ブリコレ ロック・ ・ ブリロック・ ・ ブリロック・ ・ ブリロック・ ・ ブリロッ



「構成プロパティ」-「詳細」

<mark>文字セット</mark> マルチ バイト文字セットを使用する

「C/C++」-「全般」

追加のインクルード ディレクトリ C:¥SuiteChidoriEngine¥include

デバッグ情報の形式 プログラム データベース (/Zi)

<mark>SDL チェック</mark> 空白

複数プロセッサによるコンパイル いいえ

「C/C++」-「最適化」

<mark>最適化</mark> 最適化 (速度を優先) (/Ox)

インライン関数の展開 拡張可能な関数すべて (/Ob2)

組み込み関数を使用する はい (/Oi)

<u>速度またはサイズを優先</u> 実行速度を優先 (/Ot)

プログラム全体の最適化 はい (/GL)

「C/C++」-「プリプロセッサ」

プリプロセッサの定義 ※各シンボルはセミコロンで区切って入力してください。

 $\label{eq:WIN64} WIN64; GL; _DEBUG; _WINDOWS; \% (PreprocessorDefinitions)$





「C/C++」-「コード生成」

基本ランタイム チェック 既定

「C/C++」-「言語」

準拠モード

いいえ (/permissive)



構成(C): Debug ✓ ブラットフォーム(P): x64 ✓ 構成マネージャー(O)... プリプロセッサ コード生成 出力ファイル 進行状況の表示 パージョン インクリメンタルリン \$(OutDir)\$(TargetName)\$(TargetExt 設定なし コード生成 言語 ブリコンパイル済みヘッ 出力ファイル ブラウザー信報 外部インクルード 詳細設定 すべてのオプション コマンドライン ウー いいえ (/INCREMENTAL:NO) \sim シクリメンタル リンク データベース ファイル \$(IntDir)\$(TargetNa はい (/NOLOGO) インリンクルリンチェーバーム キャー単編のサテーバーム オンの急速 コーザーごをのがイルトト ライブラがな考慮(ハットル) ライブラがな考慮(ハットル) ライブラがな考慮(ハットル) リンクステークス DL を(ハットック・クスない) リンク 大学・クンの たいした。 サービーン レージョンの高性の高速 著作権情報の非表示 いいえ いいえ いいえ C:¥Sui Engine¥lib¥x64¥Debug ソカー 全般 入力 マニフエストファイル デパッグ マンステム 景急化 埋め込み IDL Windows メタデータ 詳細形定 サイズのオブション コマンドライン > はい いいえ インクリメンタル リンクを有効にする インクリメンタル リンクを有効にします。(/INCREMENTAL, /INCREMENTAL:NO) OK キャンセル 適用(A)

「C/C++」-「プリコンパイル済みヘッダー」

プリコンパイル済みヘッダー 使用 (/Yu)

「リンカー」-「全般」

インクリメンタル リンクを有効にする いいえ (/INCREMENTAL:NO)

追加のライブラリ ディレクトリ C:¥SuiteChidoriEngine¥lib¥x64¥Debug





「リンカー」-「入力」

追加の依存ファイル ※各シンボルはセミコロンで区切って入力してください。

aoi.lib;fbx.lib;graphicsGL.lib;kernel.lib;libfbxsdk. lib;movie.lib;net.lib;peripheral.lib;pet.lib;sound.lib;zlib. lib;%(AdditionalDependencies)

「リンカー」-「システム」

サブシステム 設定なし

設定が完了したら「OK」ボタンをクリックして、画面を閉じます。

3-3 プロジェクトの設定(構成:Release、プラットフォーム:x64)



「構成:Release」「プラットフォーム:x64」になっている事を確認し、右側「ソリューションエクスプローラー」-プロジェクト名 (GameProject)を選択し、右クリックで「プロパティ(R)」をク リックします。

成(<u>C</u>): Release	 ブラットフォーム(P): 	x64	> 構成マネージャー(Q)
▲ 構成プロパティ ▲ 金融 詳細 デパッグ VC++ディレクトリ b) C/C++ ▲ リンカー 金融 入力 マニフェストファイル デパッグ	サゴンステム 量信服を更な(ジョン) とーゴ ジイスの砂定 とーゴ ショント サイスの砂定 スタックのヨント サイス スタックのヨント サイス 大きいサイスのアドレス ケーご カート 	コンソール (/SUBSYSTEM.CONSC レルス レルス	DLE)
システム 参連化 増あ込み DL Windows X9データ 詳細設定 すべてのオブション コマンドライン Þ マニフエスト ゲール Þ XML ドキュントト ジェネレー Þ ブラウナー使根 Þ ビルドイベント	サプシステム /SUBSYSTEM オプションは、オペレーティング /シルージェアドリ オプションは、オペレーティング	「システムL.exe フィールの東行方法を指定」 (またエントリーボイン) 最初 が客道を受け	.ます。サブシステムの選択によって、 ます。

<mark>構成(C)</mark> Release

成(<u>C</u>):	Release		✓ プラットフォーム(P):	x64		~	構成マネージャ	-(<u>Q</u>)
▲ 構成 1 ま ラ マ レ レ リ	プロ(ティ A 自然 単語 「バッグ C++デルクトリ /C++ レカー 全般 入力 マニフェストファイル デバッグ システム 要連 ・ デバッグ ジステム 要連 し、 Windwark 19年-9	*	共通言語シジイムサポート NETFamework 対象(ージ32) マタージドインワルジルドレドを含む 詳細プレバチィ タージトドンドルが広張子 ジーンタは、利用・Pathを加え を完まるとした・シールフーキャラシャ デバッライブンジの忠子 のムロト・ハロンジンドン・構成してき のムロト・ハロンジンドン・構成してき いたり、のンド・ディーター・キャラント のムロト・ハロンジン・シーレーキャラント のムロト・ハロンジン・シーレーキャラント のムロト・ハロンジン・シーレーキャラント のムロト・ハロンジン・シーレーキャラント のムロト・ハロンジン・シーン・キー メーレート のムロト・ハロンジン・シーン・キー メーレート メーレート マング・ムロンジン・シーレーキ ・シーン・シーレーキ ・シーン・シーン・キー ・シーン・シーン・キー ・シーン・シーン・シート ・シーン・シーン・シーン・シーン・シーン・シーン ・シーン・シーン・シーン・シーン ・シーン・シーン・シーン ・シーン・シーン ・シーン・シーン ・シーン	bにする ルのコピー	共通書語ランタイムサポートを使用し いいえ .eve *.eve	:*.ipdb;*	.iobj;*,resource	s;*.tlb
0 7 0 X 0 7 0 E	詳細設定 すべてのオブション コマンドライン にフェストッール ML ドキュメントジェネレー ラウザー情報 ルドイベント	文 : 指	MFC の使用 文学セット プログラム全体の最適化 MSVC ツールセット パージョン 学セット 定した文字セットを使用するようコンパイ	ラを設定	様率 Windows ライブラリを使用する マルチ バイト文字セットを使用する リンク時のコード生成を使用 既定 にします。ローカリゼーションで使用されま	3 .9.		~



「構成プロパティ」-「詳細」

```
<mark>文字セット</mark>
マルチ バイト文字セットを使用する
```

「C/C++」-「全般」

追加のインクルード ディレクトリ C:¥SuiteChidoriEngine¥include

SDL チェック 空白



「C/C++」-「プリプロセッサ」

プリプロセッサの定義

※各シンボルはセミコロンで区切って入力してください。

WIN64;GL;NDEBUG;_WINDOWS;%(PreprocessorDefinitio ns)

 Cunativest 201/61 A-22
 2
 X

 電気(C)
 Relase
 ブラットフォーム(D):
 x64
 運気(T = -0)*(-0)

 電気(C)
 電気(T = -0)*(-0) 電気(T = -0)*(-0) 電気(T = -0)*(-0) 電気(T = -0)*(-0)

 電気(T = -0)*(-0) 電気(T = -0)*(-0) 電気(T = -0)*(-0) 電気(T = -0)*(-0) 電気(T = -0)*(-0)

 (F)
 C(T = -0)*(-0) =
 (D)
 (D)</td

「C/C++」-「言語」

<mark>準拠モード</mark> いいえ (/permissive)

構成(C): Release			
Hereiter Hereiter	✓ ブラットフォーム(P): x64	~	構成マネ−ジャ−(<u>O</u>)
▲ 観点プロ(ケイ ▲ 金段 芽用 デパップ VC++ デルクトリ ム CC+4 金段 景通化 プリプロセッサ コーモ車成 言語	プリコンパイル法カペッチー 使用 プリコンパイルスカペッチーファイル stata プリコンパイルスチャッチー出力ファイル S(int	(/Yu) %.h Dirj\$[TargetName].pch	
ノリコンパイル(茶みへ) 出力ンデイル ブラウザー福泉 外部インクルード 詳細語定 すべてのオブション コマンドライン レ リンカー レ マニフェストッール レ XML ドキュント ジェネレー	ブリコンパイル済みヘッダー		
▶ ブラウザー情報 く >	プリコンパイル済みヘッダーの作成/使用:ビルド時にフ	ブリコンパイル済みヘッダーを作成、使用しま	\$"_ (/Yc、/Yu)



100001 プロパティベンジ ア 電気(2): Release ブラウトフオーム(2): x64 電気マン(2): 電気(2): 電気(2): アへての気ならってグン(2): 第 アパッグ マムマ・ディングトン アへての気ならってグン(2): 第 アパッグ マムマ・ディングトン アムマ・クシャー(2): 10: アメン アン </t

Table20147501741179
 Toph734-6UE: x64
 State
 State
 Toph734-6UE: x64
 State
 St

「C/C++」-「プリコンパイル済みヘッダー」

プリコンパイル済みヘッダー 使用 (/Yu)

「リンカー」-「全般」

追加のライブラリ ディレクトリ C:¥SuiteChidoriEngine¥lib¥x64¥Release

「リンカー」-「入力」

追加の依存ファイル ※各シンボルはセミコロンで区切って入力してください。

aoi.lib;fbx.lib;graphicsGL.lib;kernel.lib;libfbxsdk. lib;movie.lib;net.lib;peripheral.lib;pet.lib;sound.lib;zlib. lib;%(AdditionalDependencies)

「リンカー」-「システム」

サブシステム 設定なし



3-4 プロジェクトのビルド

and the second second						_
S (and sign						
a data and				1 100	-140 111/0-1- 080 Odd	
Same Same at					172-310 Canehojart (M 70/27)	
				· · · · · · · · · · · · · · · · · · ·	00 140 200 163-16019-275732 1627-686 9271-686 9271-686 9271-686 1000-1000-171-077	
				Ø 84	10,13-7980990() 99/13-340 173/10-5-003-00	
				CL/	CONTAC	
		3 BUDBESS.	OH-SHR-A	4.0	10	
		5 BEACHSP-(F)	part-ar-a	1 No	(1777-122重用N-	Carrys
		#5.0.		0 37-	+7+770327+1888/H	
		9 7530.		X er	drace.	Out-X
		8 Bill Machine		× eb	00	0el
				700 700 700 910 910 910	103200 12740202-00 127402880986028428 127408988579-88088428 1490088284200 19-54-322-888 19-54-322-888	2
				C 177	1万2-5-72×1-5-8第120 (257第5	
		05 :	281 97 0	1 1 70	(9-cm)	At-Inte
1.0				×		
105 - 01 40 0 0 23 25	- 6 N N Z W	63 :	27.1	*7 0	17 00 p 70 + 1 K	D F-04286 77 (4) # 7809-08
				(6.6 70.7		

新しい項目の追加 - Gai								×
▲ インストール済み		並べ替別	; 既定	- IF 🗉		検索 (Ctrl+E)		p.
J−F		Ľï	C++ ファイル (.cpp)		Visual C++	種類: Visual C++	金ポファイルを作成	1.23
書式設定 データ		ĥ	ヘッダー ファイル (.h)		Visual C++	0		
リソース Web		15	C++ クラス		Visual C++			
ユーティリティ プロパティ シート		-	C++ モジュール インターフェイス	ユニット (.ixx)	Visual C++			
HLSL グラフィックス								
▶ オンライン								
名前(N):	main.cpp							
場所(L):	C:¥SuiteChidoriEngir	re¥Gamel	roject¥src¥		-	参照(<u>B</u>)		
							直加(A) ++	ッンセル



「リンカー」-「詳細設定」

安全な例外ハンドラーを含むイメージ いいえ (/SAFESEH:NO)

設定が完了したら「OK」ボタンをクリックして、画面を閉じます。

右側「ソリューションエクスプローラー」-プロジェクト名 (GameProject)を選択し、右クリックで「追加(D)」-「新しい項 目(W)」をクリックします。

「新しい項目の追加」画面が開くので、設定を行います。

ファイル

C++ ファイル (.cpp)

名前

main.cpp

場所

C:¥SuiteChidoriEngine¥GameProject¥src¥ (先ほどプロジェクト下に作った「src」フォルダを選択します)

「追加(A)」 をクリックします。

右側「ソリューションエクスプローラー」-「main.cpp」をクリックし、作業ディレクトリに下記コードを入力します。

/**

main.cpp */

7

#include "stdafx.h"
int WINAPI WinMain(_In_ HINSTANCE hInstance,

_____In_opt_ HINSTANCE

hPrevInstance,

_In_LPSTR lpCmdLine, _In_ int nCmdShow)

{ MessageBox(NULL, TEXT("Welcome Suite Chidori Engine!"),

TEXT("メッセージボックス"), MB_OK | MB_ ICONINFORMATION);

- return 0;
- }

Scanehoject + (70-7/5/23-7) +	
Constant (Constant) (Cons	(07) 07(17) - 1 4 4 5 (00) (07) 07(17) - 1 4 5 (00) (07) 07(17) (06) 07(17) 07(17) (06)
	VIII-LAV TOTING A ST
	ソリューション 1937日-5- Gr 家王 プロパティー・キン
	ソリューション 19270-5- Ge 変更 プロパティー・キッ
	992-942/2020-9- Gr E E 75577
at 62 45 4-4 (54 281 195 00	yys-ysy123270-5- Grgg 7007γ • ₽ 3 2017β ≠
as 61 ¥5 € - 9 (34 221 12C 00	V92-942 123/30-9- 64 KR 70:07 • • • • • •
нь 0 г. Ал 0 + э — — — — — — — — — — — — — — — — — —	910-92 1325-5- 64 KK 7004
nt- 92 kt 6-9 85 get 65 00 22 2200g - €5558 10 19	YUL-YO 1333-5- GKR 7007、 ・ ₹ 副本 タ
ns.⊂ 61 41 6 6 15 15500	99-00 1178-9- 0481 700

second a state of a state of	1	• • VID-14/21120-3-	
Salaran Booked Teacher	- Decision Arca	2. Mil-bio period dia manadamentary assessed paradamentary second strategy and a second se	2 8 10 P
Control of the second se	santana. Suite dhalari (symm ⁽¹), .el.jok el.jtokimilaritoki:	▲ Dives 2016 2016 2016 8413-14019-0273 8413-14019-0273	10
a l'		70/371-646-01 70/370-646-01 10/370-01 10/370-010	014-227
		B #0.071-90-1713	0-1-8(3-N
		(LI-CO-MEAL)	
		■ 10 ³ 1933 1979-1922. ■ Nacher/197-19日間の	Qr+94
		38->7-y7 750/07+22 Kit-y500	124
		X statuto S surveys X statuto	CM-X CM-V BM
		***********************************	кодици
		C 23330-9-23669-	and the second sec
N		01 29-0 5K 01 # 100-00	and one
		+12	
0.8.0	E 5.5 11 N		
		10	

							_
	anop a x Marcop Marca	1		An Instance (1973) and a st	instanting in	201-240 11120-3-	
	1 100					0000 0.000 1	a
						The second second second	-
	4 Birclude Middle A					A N (and had	-
	Ja.met, KONITABLE hPresidentance,					E CLEM	
	a la matica)					002400	
1 1	Bestandte/MAL, 2017 Christen Saite Chilleri Ergine"	2.				77-276	
	1日11(アノッセージボックスで)。10_0K 10_00(M100	6/100bs				E3-040	
1 1						1112 J-1013-777720	
						This Part and the Part of the	
						174733-7874A0	
						B BLUTTA-140 17370-5-0(1-N	
A 0						CLFOOR BEAR	
No. 0.0 1.0 0.0 1.0 0.0 1.0 0.0 <td></td> <td></td> <td></td> <td></td> <td></td> <td>450</td> <td></td>						450	
					5	 クリスクイデードロー NuGerパックージの変更Nー 	Col-Sult-
1 010					•	3オートアッププロジェクトに配置(4) ドバック(6)	
A1 A1<						E esteva	Col+X
A 1 4 1 4 0 0 4125 4 0 0 4125 4 0 A 1 4 1 4 0 0 4125 4 0 0 4125 4 0 0 4125 4 0 0 4125 4 0 0 4125 4 0 0 4125 4 0 0 4125 4 0 0 4125 4 0 0 4125 4 0 0 4125 4 0 0 4125 4 0 0 4125 4 0 0 4125 4 0 0 4125 4 0 0 0 1125 4 0 0 0 1125 4 0 0 0 1125 4 0 0 0 1125 4 0 0 0 1125 4 0 0 0 1125 4 0 0 0 1125 4 0 0 0 1125 4 0 0 0 1125 4 0 0 0 1125 4 0 0 0 1125 4 0 0 0 1125 4 0 0 0 1125 4 0 0 0 1125 4 0 0 0 0 1125 4 0 0 0 1125 4 0 0 0 0 0 0 0 0 0 <						< whoe	0wl
1 1						D 48082M	12
No. Old 10 Other State (State						プロンとうトロアンロード(1) プロンとうトロ亜東京市学校会団あらしみ プロンとうトロ亜東京市学校会団あらしみ プロンシンロロ構計をいう一生きの読みらしみ ッピューシンロロ構計をいう(1) 参加データドー215ーも長本 参加データドー215ーも長本	
10 11<						1222/0-5-22x6/-6800	
	15 7 82 48 8 9			67 39	N PK (N	* 20/1+m	Abulater
Marrie Marrie<							
12-2 88 720.0 2-1 8 0 0.05 0.05 0.05 0.05 0.05 0 0.05 0.05 0.05 0.05 0.05 0 0.05 0.05 0.05 0.05 0.05	/12-540/88 · Q315+ ▲488 @4.618-0 🎙 (5.6+)	meldene +		150198	. م		
0.000 Des PR121-2882422, walk-width Old BRURENSUL Berlings Reven 1 0.000 Des PR121-2882422, walk-width Old BRURENSUL Berlings Reven 1 0.000 Reven Berlings Reven 1	12-F RR		70%29	2164	e		
B. Imi: 2015/0129-888/20128-8966,0268,06000 General sectors Constraints of the sector of th	A LON FOR B& THELS-FRELELS, Intelline FOR THE BERGE	84768.	Ganahojan	main.opp			
V 100 Note (P120EEEL, Not 75 and N p Sector) Gardigat MAA op 4 A	A DIME 27/F 5/2 IS-I REGISTER WINH GLIODERG, WHOWS		Genefisjent	main.opp			
(42)	COD Include 2715 (2012) 23. Weak 3 the such file or directory		Ganahojat	intervicep			
						(6.8)	

 プラットフォーム x64

右側「ソリューションエクスプローラー」-プロジェクト名 (GameProject)を選択し、右クリックして「ビルド(U)」をクリッ クします。

ソリューション構成を「Release」に切り替え、同じくプロジェ クト(GameProject)を選択し、右クリックして「ビルド(U)」をク リックします。

ビルドが成功すると、「プロパティ(R)」-「構成プロパティ」-「全般」-「出力ディレクトリ」が示すフォルダに.exeファイルが 生成されます。



生成された.exeファイルと同じディレクトリに「libfbxsdk.dll」 ファイルを追加します。

※Debug・Releaseどちらにも追加してください
※libfbxsdk.dllは、(インストール先のフォルダ)\lib\dll\x64の中
にあります。

「Visual Stusdio 2019」 画面で「F5」を押して実行し、 「Welcome Suite Chidori Engine!」 のダイアログが表示され れば成功です。



第3章 操作ガイダンス(スイート千鳥エンジンエディター)

1 事前準備

1-1 ライセンス認証

SuiteChidoriEngineTool						
ファイル ホーム 共有	表示					
★ 日本 100000000000000000000000000000000000	★ 切り取り ■ パスのコピー ♪ ショートカットの貼り付け		 ・ 名前の 新し 変更 7オル 	「自新しいアイテム・ 」 うヨートカット・ 、 、 ダー	 ブロバティ 第 第 ブロバティ 2 2 2 2 2 2 2 2 3 2 3 3 4 3 4 5 <l< th=""><th> ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・</th></l<>	 ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・
クリップボ	-F	整理		新規	聞く	選択
🖓 📙 🗢						
← → ~ ↑ □ > PC >	ローカル ディスク (C:) → Su	iteChidoriEngine > tools	→ Windows →	SuiteChidoriEngineTo	ol	
	名前	· · · · · ·	更新日時	種類	サイズ	
x 9199 792X	sampleData		2022/10/13 11:17	ファイル フォル	<i>4</i> -	
o Creative Cloud Files	shaders		2022/10/13 11:17	ファイル フォル	<i>4</i> -	
 OneDrive 	🙈 AppBoot.exe		2022/09/16 9:41	アプリケーショ	ン 3,3:	23 KB
- Onebine	Chidori.ls		2022/11/02 11:58 LS ファイル	1,13	26 KB	
PC PC	ChidoriTool.exe		2022/09/27 11:51	アプリケーショ	ソ 31	73 KB
- USB 65/71(D)	libfbxsdk.dll		2021/04/08 14:34	アプリケーショ	ン拡張 8,21	38 KB
036 P91 7 (C)	😵 SuiteChidoriEnginel	Help.chm	2022/09/28 11:48	コンパイルされ	ルたHT 27,5!	56 KB
> 💣 ネットワーク	SuiteChidoriEngine	foolLib.dll	2022/09/27 11:40	アプリケーショ	ン拡張 3,95	77 KB

アクティベーションファイル(Chidori.ls)を「(インストール先)\ tools\Windows\SuiteChidoriEngineTool」に設置します。

「ChidoriTool.exe」をダブルクリックします。

SuiteChidoriEngineTool	管理					
ファイル ホーム 共有	表示 アプリケーション う	1-11,				
クイックアクセス コピー 貼り付け にどう留めする	★ 切り取り ■ パスのコピー ▶ ショートカットの貼り付け	▲ 修動先 コビーチ	 4前の 変更 フォルダー 	「計算」が、「おけいアイテム▼ 「計算」とヨートカット▼	✓	計 すべて選択 計 選択解除 計 選択の切り替え
クリップボ	-6	整理		新規	麗く	選択
2 📙 👻						
← → ~ ↑ □ → PC :	> ローカルディスク(C:) > Su 名前	iiteChidoriEngine → tools	→ Windows → Si 更新日時	uiteChidoriEngineTool 種類	サイズ	
📌 ウイック アクセス	sampleData		2022/10/13 11:17	79411.7+11.47-		
Greative Cloud Files	shaders		2022/10/13 11:17	ファイルフォルダ	-	
:	AppBoot.exe		2022/09/16 9:41	アプリケーション	3,323 KB	
 OneDrive 	Chidori.ls		2022/11/02 11:58	LS 7711	1,126 KB	
PC	ChidoriTool.exe		2022/09/27 11:51	アプリケーション	373 KB	
une l'excelure	libfbxsdk.dll		2021/04/08 14:34	アプリケーション打	t張 8,288 KB	
- USB P71 / (E)	😵 SuiteChidoriEngine	Help.chm	2022/09/28 11:48	コンパイルされた	HT 27,556 KB	



Cancel

シリアルコードを入力するダイアログが表示されるので、入力 して「OK」 ボタンをクリックします。 シリアルコードはハイフンも含めて入力してください。 ライセンスが正しく認証されると、スイート千鳥エンジンエディ ターが表示されます。

認証に失敗する場合は、アクティベーションファイル(Chidori. ls)が正しく設置されているかと、シリアルコードが正しく入力さ れているかを確認してください。

1-2 アプリサイズの設定

W 📲 🖬 📲 💼 💼 👘

認証



アプリサイズを変更する場合は、メニュー「設定」→「アプリサ イズ」から、アプリサイズダイアログを表示して、サイズを指定 します。

ここで入力したアプリサイズは、スイート千鳥エンジンエディ ターを再度起動するまでは反映されないので、ご注意くださ い。

2 プロジェクト作成



アクティベーションファイル(Chidori.ls)を「(インストール先)\ tools\Windows\SuiteChidoriEngineTool」に設置します。



プロジェクトを作成すると、シーンと透視投影カメラ、ディレク ショナルライト、アセットフォルダが1つずつ追加されます。 オブジェクトリストに表示されるプロジェクト、シーン、カメラ、 ライトをクリックして選択することで、選択した要素のプロパ ティがプロパティリストに表示されます。



シーンの背景色を変更します。 オブジェクトリストから「シーン_1」をクリックして、プロパティ リストにシーンのプロパティを表示します。 「背景色」の「緑」の値を「150」に変更して、アプリの背景色 を緑色にします。

3 モデルの追加



モデルを追加するために、まずモデルデータを追加します。 データリスト内の「data」をクリックして選択します。

「追加」ボタンをクリックして表示されるリストから、 「モデル」を選択して、「(インストール先)\tools\Windows\ SuiteChidoriEngineTool\sampleData\minidragon.fbx」を指定 します。

指定したモデルデータがデータリストに表示されます。



E7/6_1

8**1**6

モデルオブジェクトを追加します。 オブジェクトリストの「シーン_1」の右クリックメニューから「追 加」→「モデル」を選択します。モデルオブジェクト「モデル_1」 がシーンの子要素に追加されます。

オブジェクトリストの「モデル_1」をクリックして、プロパティリ ストにモデルのプロパティを表示します。

「モデルデータ」のコンボボックスをクリックすると、追加した モデルデータ「minidragon.fbx」が表示されるのでクリックし て選択します。これで選択したモデルがシーンビューに表示さ れます。



モデルの位置とサイズ、向きを調整します。

モデルの位置とサイズ、向きは、モデルが持つトランスフォーム という部品に対して適用されますが、適用するトランスフォー ムが正しくないと、モデルが崩れたり、変更が適用されなかっ たりします。

今回読み込んだモデルでは、「Dragon01」というトランス フォームを使用するとよいので、「変形対象のトランスフォー ム」内の「Dragon01」にチェックを入れます。チェックを入れ るとモデルの大きさが変化しますが、これはモデルに元々設定 されていた値ではなく、モデルオブジェクトの初期値が反映さ れるためです。

「位置」の「Y」に「-150.0」を、「スケール」の「X」「Y」「Z」それぞれに「20.0」を、「回転」の「Y」に「45.0」を入力して、モデルの位置とサイズを調整し、右方向に向かせます。

4 モーションの設定

4

ーン_1 透明投影カメラ_1 ディレクショナルライ モデル_1

18.10



モーションを設定するために、まずモーションデータを追加します。

データリスト内の「data」をクリックして選択します。

「追加」ボタンをクリックして表示されるリストから「モー ション」を選択して、「(インストール先)\tools\Windows\ SuiteChidoriEngineTool\sampleData\minidragon_motion. fbx」を指定します。

指定したモーションデータがデータリストに表示されます。



モーションオブジェクトを追加します。 オブジェクトリストの「モデル_1」の右クリックメニューから 「追加」→「モーション」を選択します。モーションオブジェクト 「モーション_1」がモデルの子要素に追加されます。



オブジェクトリストの「モーション_1」をクリックして、プロパティリストにモーションのプロパティを表示します。

「モーションデータ」のコンボボックスをクリックすると、追加 したモーションデータ「minidragon_motion.fbx」が表示され るのでクリックして選択します。

なお、モーションはエディター上では再生されず、プレイモード という画面で再生されます。そのため、この時点ではまだモデ ルのモーションは再生されません。



モーションをループ再生するようにします。 「ループ」のコンボボックスをクリックして「ループする」 に変 更します。

5 アプリ作成



アプリを作成する前に、今回作成したシーンの表示を確認しま す。

メニュー「ビルド」→「プレイモード」からプレイモード画面を 表示します。

プレイモード画面では、エディター上では再生されないモデル のアニメーションやパーティクルエフェクト、サウンド、ムービー の再生を確認することができます。





プレイモード画面では、マウス操作による視点変更が可能で す。 プレイモードを終了するには、プレイモード画面をばつボタン で閉じます。

プレイモードで表示を確認したら、アプリを作成するために Visual Studioのソリューションファイルを出力します。 メニュー「ビルド」→「出力」から出力設定ダイアログを表示し

ます。 ダイアログで各設定を入力します。

「作業フォルダ」はソリューションファイルの作成時に一時的 に使用するフォルダです。

ここでは例として「(インストール先)\tools\Windows\ SuiteChidoriEngineTool\output\temp」を指定します (インストール先がデフォルトフォルダの場合は「C:\ SuiteChidoriEngine\tools\Windows\SuiteChidoriEngineTool\ output\temp」)。

「プロジェクト出力先フォルダ」にはソリューションファイルー 式を出力するフォルダを指定します。

ここでは例として「(インストール先)\tools\Windows\ SuiteChidoriEngineTool\output」を指定します(インストール先 がデフォルトフォルダの場合は「C:\SuiteChidoriEngine\tools\ Windows\SuiteChidoriEngineTool\output」)。

「ライブラリの一覧」にはアプリで使用するライブラリが設置 されたフォルダを指定します。

「(インストール先)\lib\tool」を指定してください(インストール 先がデフォルトフォルダの場合は「C:\SuiteChidoriEngine\lib\ tool」)。



「出力」ボタンをクリックすると、「プロジェクト出力先フォル ダ」にフォルダが生成され、ソリューションファイルー式が出力 されます。

生成されたフォルダ内にプロジェクト名のフォルダ (ここでは 「プロジェクト」フォルダ)があり、その中にプロジェクト名の ソリューションファイル(ここでは「プロジェクト.sln」)がありま す。



ソリューションファイルをダブルクリックします。 Visual Studioが起動したら、「ソリューション構成」を 「Release」にして「開始」をクリックします。 ウインドウが起動し、作成したアプリが表示されます。



1ル ホーム 共有	表示 アノリケーション フール			~
-> * 🛧 📙 « out	tput > プロジェクト > プロジェクト > bin >	Release	ר ער אין	eleaseの検索
+ 0/~0 70h7	名前	日付時刻	種類	97 J
× 9199 792X	minidragon body tsn.dds	2021/01/07 11:51	paint.net Image	21,846 KB
OneDrive - Personal	minidragon_eye_col.dds	2021/01/07 11:51	paint.net Image	683 KB
PC .	minidragon_eyeshadow_col.dds	2021/01/07 11:51	paint.net Image	1,366 KB
	minidragon_motion.fbx	2021/01/07 11:51	3D Object	7,898 KB
🧈 ネットワーク	Minidragon_nails_col.dds	2021/01/07 11:51	paint.net Image	683 KB
	🛃 minidragon_nails_tsn.dds	2021/01/07 11:51	paint.net Image	5,462 KB
	minidragon_tooth_col.dds	2021/01/07 11:51	paint.net Image	683 KB
	minidragon_tooth_tsn.dds	2021/01/07 11:51	paint.net Image	5,462 KB
	SuiteChidoriEngineTooll ib.dll	2022/09/13 11:54	アプリケーション拡張	5.647.KB
	■ プロジェクト.exe	2022/09/15 15:46	アプリケーション	10 KB
	」 ノロジェクト.exe.config	2022/09/15 15:45	Embarcadero KAD	1 KB
	前 プロジェクト.pdb	2022/09/15 15:46	Program Debug D	38 KB

アプリの実行ファイルは、「(プロジェクト出力先フォルダ)/(プ ロジェクト名フォルダ)/(プロジェクト名フォルダ)/bin/Release」 フォルダに出力されています。 アプリを配布する場合は、フォルダ内のファイルを全て含めて ください。

第4章 Q&A

1 スイート千鳥エンジンについて

Q1-1 スイート千鳥エンジンとは何ですか?

A1-1 スイート千鳥エンジンは、プログラミングソフトと組み合わせて使用するライブラリ集(ソースコード集)となります。

Q1-2 スイート千鳥エンジンでは、どのプラットフォーム向けのゲームが作成できますか?

- A1-2 WindowsとiOSになります。なお、インストーラはWindows用ゲームを作成するためにWindowsで使用できるWindows版 と、iOS用ゲームを作成するためにmacOSで使用できるiOS版とがあります。
- Q1-3 スイート千鳥エンジンで使用するプログラミング言語は何ですか?
- A1-3 スイート千鳥エンジンで使用するプログラミング言語はC++です。

2 導入について

Q2-1 スイート千鳥エンジンのインストーラについて教えてください。

Windows版とiOS版の両方のバージョンのインストーラを提供しています。使用するプラットフォームごとに、下記の通りイ ンストールしてください。

【Windows版】

Windowsで、「setup_スイート千鳥エンジン_Windows.exe」または「スイート千鳥エンジン_Windows.msi」を起動してインストールしてください。

【iOS版】

macOSで、「SuiteChidoriEngine_iOS.pkg」を起動してインストールしてください。

警告メッセージが表示され起動できない場合は、「SuiteChidoriEngine_iOS.pkg」を選択し、右クリックメニューの「開く」から起動すると、 表示される警告メッセージ内に「開く」ボタンが表示されますので、そのボタンから起動してください。

Q2-2 インストール後、まず何をすればよいですか?

A2-2

A2-1

スイート千鳥エンジンをインストールした場所にSuiteChidoriEngineというフォルダが作成されます。 デフォルト設定では、 Windows版は「C:\SuiteChidoriEngine」 iOS版は「/Application/FORUM8/SuiteChidoriEngine」 スイート千鳥エンジンヘルプの [チュートリアル] を参考にして、スイート千鳥エンジンの使い方を確認してください。 ライセ ンス認証の手順についてもここで学ぶことができます。

Q2-3 必要なプログラミングソフトは何ですか?

A2-3 使用するプラットフォームごとに、下記の通りとなります。

【Windows版】

Windows版はVisual Studio 2019を使用します。下記のマイクロソフトのダウンロードサイトより入手し、インストールして ください。

https://docs.microsoft.com/ja-jp/visualstudio/releases/2019/release-notes

インストール時には、「C++ によるデスクトップ開発」 にチェックを入れてください。 なお、Visual Studio 2019の全てのエディションにて動作しますが、使用にあたっては該当するエディションのライセンス 条項をご確認下さい。また、個人利用の場合はコミュニティ版が無償で使用できます。 コミュニティ版は上記URL内の 「Download Commuity 2019」 ボタンからダウンロードできます。

【iOS版】 iOS版はXcodeを使用します。App StoreよりXcodeをインストールしてください

Q2-4 ヘルプファイルはどこで確認できますか?

A2-4 スイート千鳥エンジンヘルプは、以下より確認できます。

【Windows版】

Windowsメニュー→ [FORUM8] → [スイート千鳥エンジン ヘルプ]

【iOS版】

 $[\mathcal{P}\mathcal{P} \cup \mathcal{P} = \mathcal{P}]$ → [FORUM8] $\mathcal{P} \cup \mathcal{P}$ [SuiteChidoriEngine] $\mathcal{P} \cup \mathcal{P} \cup \mathcal{P}$ [document] $\mathcal{P} \cup \mathcal{P} \cup \mathcal{P} \cup \mathcal{P} \cup \mathcal{P}$ [iOS] $\mathcal{P} \cup \mathcal{P} \cup \mathcal{P$

Q2-5 サンプルデータはありますか?

A2-5 サンプルプロジェクト、および、スイート千鳥エンジンで作成したゲームを用意しております。

【Windows版】 ・チュートリアル モデルの表示とアニメーション、サウンド機能を確認できるサンプルです。スイート千鳥エンジンヘルプの [チュートリア ル] を確認してください。 ・ParticleSample パーティクルエフェクト機能を確認できるサンプルです。スイート千鳥エンジンヘルプの [サンプル | ParticleSampleの使 用方法] を確認してください。 ・PostEffectSample ポストエフェクト機能を確認できるサンプルです。スイート千鳥エンジンヘルプの [サンプル | PostEffectSampleの使用方 法] を確認してください。 ・ゲーム スイート千鳥エンジンで作成したゲームです。スイート千鳥エンジンヘルプの [ゲーム] を確認してください。

【iOS版】 ・チュートリアル モデルの表示とアニメーション、サウンド機能を確認できるサンプルです。スイート千鳥エンジンヘルプの [チュートリア ル] を確認してください。

3 ライセンス認証について

Q3-1 発行されたアクティベーションファイル(Chidori.ls)とシリアルコードは、どのように使用しますか?

A3-1 アクティベーションファイルとシリアルコードは、ライセンス認証に使用します。 使用方法は、スイート千鳥エンジンヘルプ の [概要|ライセンス認証の手順] および [チュートリアル|Tutorialの使用方法] (Windows版)、[チュートリアル|チュート リアルの使用方法] (iOS版)を確認してください。 ライセンスについては、[概要|ライセンスについて] を確認してください。

なお、ライセンス認証を行わない場合は、正しくゲーム画面が表示されず、下記のようになります。

【Windows版】 ゲーム画面にスイート千鳥エンジンのロゴが常時表示されます。

【iOS版】 ゲーム画面が黒塗りとなり何も表示されません。

Q3-2 認証用の関数を呼んでいますが、認証が失敗してしまいます。

A3-2 グラフィックスの初期化より前に認証用の関数を呼んだ場合は、認証に失敗します。グラフィックスの初期化より後に認 証用の関数を呼んでいるかを確認してください。なお、グラフィックスの初期化については、スイート千鳥エンジンヘルプの [チュートリアル|5. フレームワーク処理の詳細説明]の5.2.3.3(Windows版)、[チュートリアル|チュートリアルの構成] の2.2.5(iOS版)を確認してください。

4 スイート千鳥エンジンの機能について

Q4-1 スイート千鳥エンジンにはどのような機能がありますか?

A4-1 スイート千鳥エンジンにはグラフィック機能やファイルの入出力機能、タスク制御機能などがあります。主要な機能やそれ ぞれの使用方法については、スイート千鳥エンジンヘルプの [機能説明] を確認してください。

Q4-2 スイート千鳥エンジンにはどのようなクラスや関数がありますか?

A4-2 スイート千鳥エンジンで使用できるクラス、関数の一覧を記述したリファレンスマニュアルを用意しています。リファレンス マニュアルは以下より確認できます。

【Windows版】

Windowsメニュー→ [FORUM8] → [スイート千鳥エンジン リファレンスマニュアル]

【iOS版】

[アプリケーション] → [FORUM8] フォルダ→ [SuiteChidoriEngine] フォルダ→ [document] フォルダ→ [iOS] フォルダ → [SuiteChidoriEngineReference] フォルダ→ [index.html]

5 スイート千鳥エンジンが対応しているファイル形式について

Q5-1 スイート千鳥エンジンが対応しているモデルデータのファイル形式は何ですか?

- A5-1 スイート千鳥エンジンが対応しているモデルデータのファイル形式はFBX形式です。テクスチャは、Windows版ではDDS 形式を、iOS版ではPVR形式をサポートしています。
- Q5-2 スイート千鳥エンジンが対応しているアニメーションデータのファイル形式は何ですか?
- A5-2 スイート千鳥エンジンが対応しているアニメーションデータのファイル形式はFBX形式です。
- Q5-3 スイート千鳥エンジンが対応しているサウンドデータのファイル形式は何ですか?
- A5-3 スイート千鳥エンジンが対応しているサウンドデータのファイル形式はWave形式とOgg形式です。
- Q5-4 スイート千鳥エンジンが対応しているムービーデータのファイル形式は何ですか?
- A5-4 スイート千鳥エンジンのWindows版が対応しているムービーデータのファイル形式はAVI形式で、映像はMotion JPEG、 音声はPCMに対応しています。また、任意のムービーデータをこの形式に変換できる動画変換ツール「MJPEGコンバー ター」を用意しています。MJPEGコンバーターについては、スイート千鳥エンジンヘルプの [ツールの使用方法]動画変換 ツールの使用方法] を参照してください。

6 その他

A6-3

- Q6-1 今まで非商用で使っていましたが作ったゲームを販売したくなりました。どうすればよいでしょうか?
- A6-1 商用利用となる場合は、有償となります。大変お手数ですが、弊社営業までご連絡をお願いいたします。
- Q6-2 スイート千鳥エンジンで作ったゲームを販売した際、売り上げに応じてお金を支払う必要はあるのでしょうか?
- A6-2 いいえ、売り上げに応じた支払は必要ありません。
- Q6-3 スイート千鳥エンジンで作ったゲームを販売している間、常にスイート千鳥エンジンのサブスクリプションを継続する必要 はあるのでしょうか?
- いいえ、 サブスクリプションの継続はゲームの開発中のみでよいです。ただ、ゲームのアップデートや修正の予定がある場 合は、スイート千鳥エンジンの最新版をご利用いただけるよう継続いただくことをお勧めいたします。
- Q&Aはホームページ(https://www.forum8.co.jp/faq/win/chidori-qa.htm)にも記載しております。

スイート千鳥エンジン Ver.2 操作ガイダンス

2022年 11月 第1版

発行元 株式会社フォーラムエイト 〒108-6021 東京都港区港南2-15-1 品川インターシティA棟21F TEL 03-6894-1888

禁複製

お問い合わせについて

本製品及び本書について、ご不明な点がございましたら、弊社、「サポート窓口」へお 問い合わせ下さい。 なお、ホームページでは、Q&Aを掲載しております。こちらもご利用下さい。

> ホームページ www.forum8.co.jp サポート窓口 ic@forum8.co.jp FAX 0985-55-3027

スイート千鳥エンジン Ver.2 操作ガイダンス

www.forum8.co.jp