

Driving Simulator Platform



Sirius

User's Manual

August, 2020

MISAKI
design.

Misaki design LLC

目次

Operation Manual	2
------------------	---

Technical Manual

1. 基本コンセプト	7
------------	---

1.1 システム・コンフィギュレーション	9
----------------------	---

1.2 プロジェクト	12
------------	----

1.3 PMTools	13
-------------	----

2. フォルダ構成	15
-----------	----

3. Score

3.1 GUI	17
---------	----

3.2 Score関連ファイル	18
-----------------	----

3.3 システム・コンフィギュレーションの編集	19
-------------------------	----

3.4 プロジェクトの編集	28
---------------	----

3.5 他PCへのログイン情報の編集	30
--------------------	----

3.6 プリファレンス設定	32
---------------	----

4. S-Interface	33
----------------	----

5. Re:simとの連携	37
---------------	----

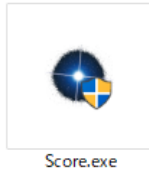
6. UE4との連携	38
------------	----

7. Function Extender	44
----------------------	----

Operation Manual

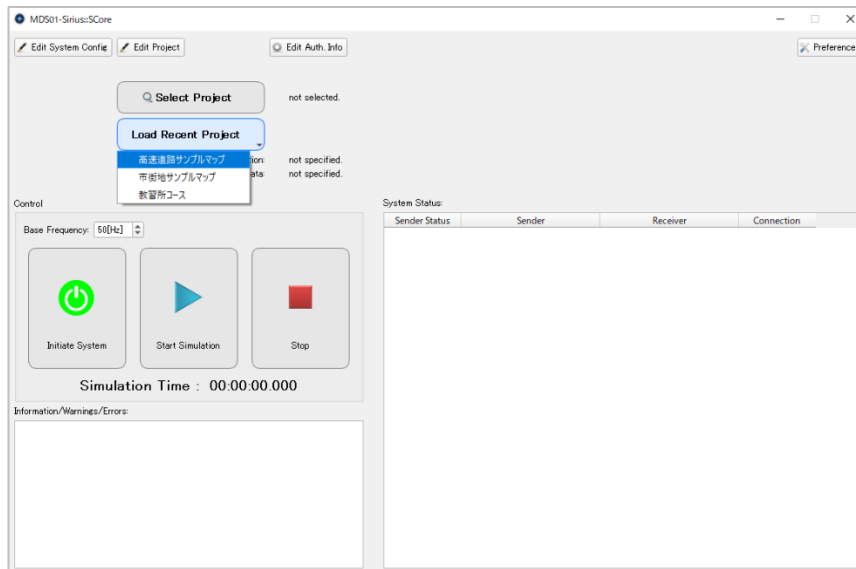
すでにシミュレーション用データが全て設定されている場合には、操作は非常に簡単です。

- ① Scoreを立ち上げます。



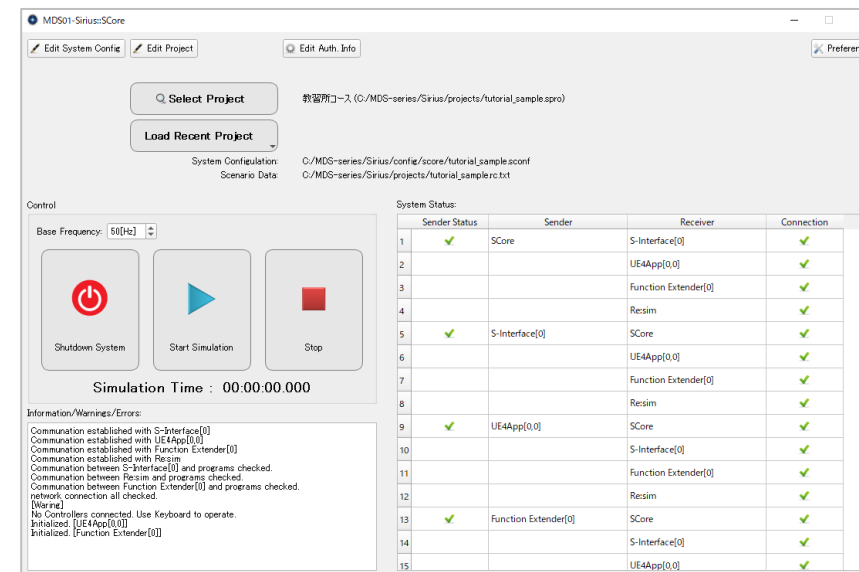
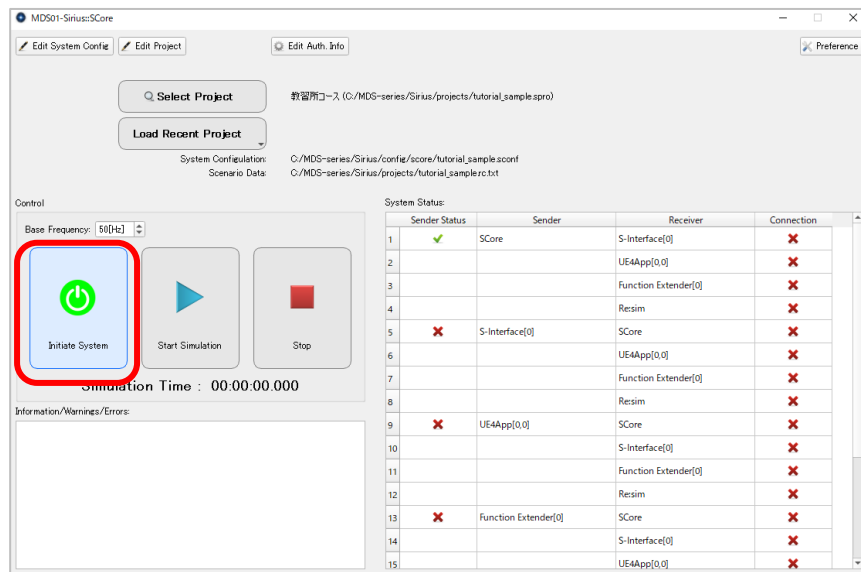
※ Scoreは管理者権限で実行する必要があります。

- ② 実行するプロジェクトを”Load Recent Project”から、もしくは”Select Project”で選択します。



※ プロジェクトの概念ならびに作成方法については、Technical Manualパートを参照のこと。
※ 左の図中のプロジェクト・リストはサンプルであり、必ずしもこの通りであるとは限りません。

- ③ “Initiate System” ボタンを押します。
または、“F1”キーを押すことでも、同様の操作ができます。



必要なプログラムが立ち上がり、通信の確認と初期化が行われます。
準備が整うと、Status, Connectionの×が✓に変わり、また、メッセージ・
ウィンドウやコンソールにも、“network connection all checked”
や”Initialized.[*]”などと表示されます。

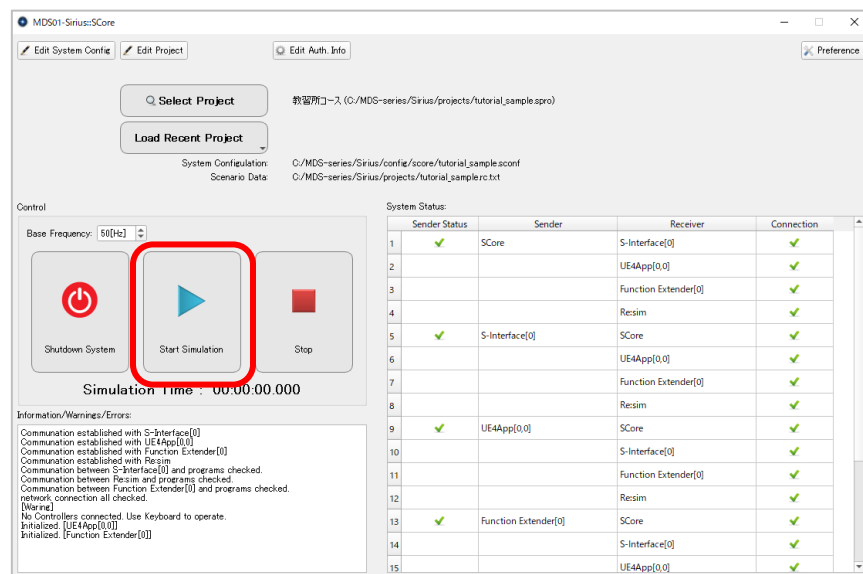
```
Score Console
sock = "Function Extender[0]" targetSock = "Function Extender[0]"
okList: append "UE4App[0,0]"
result = true
emit GetResponse
get response. hasResponse = true [nTry= 0 ]

[InitiateSystemClicked] network connection checked.
....
Send Initialization Signal to Programs.
[ 1 / 4 ] to "S-Interface[0]" : hz = 50
sock = "S-Interface[0]" targetSock = "S-Interface[0]"
result = true
emit GetResponse
"Initialized. [S-Interface[0]]"
[Warning] No Controllers connected. Use Keyboard to operate.
[ 2 / 4 ] to "UE4App[0,0]" : hz = 50
sock = "UE4App[0,0]" targetSock = "UE4App[0,0]"
result = true
emit GetResponse
"Initialized. [UE4App[0,0]]"
[ 3 / 4 ] to "Function Extender[0]" : hz = 50
sock = "Function Extender[0]" targetSock = "Function Extender[0]"
result = true
emit GetResponse
"Initialized. [Function Extender[0]]"
[ 4 / 4 ] to "Resim" : hz = 50
```

④ “Start Simulation”ボタンを押して、シミュレーションを開始します。

または、“F5”キーを押すことで、同様の操作ができます。

※ファンクションキーによる操作は、1台のPCで実行するときに、Scoreがグラフィック画面により隠れてしまい、ボタンを押せない場合に必須になります。



⑤ 終了させるときは、“Stop”ボタンを押して、その後、“Shutdown System”ボタンを押します。

または“F9”キーを押すことで、同様の操作ができます。

※開始時と同じく、ファンクションキーによる操作は、1台のPCで実行するときに、Scoreがグラフィック画面により隠れてしまい、ボタンを押せない場合に必須になります。

⑥ 再度実行する場合は、②もしくは③から手順を繰り返します。

Technical Manual

1. 基本コンセプト

Siriusは、複数のプログラムが通信によりデータをやり取りしながら、連携して動作することで、ドライビング・シミュレータの機能を実現します。

シミュレーションの実行管理を司るのは、Score(スコア)というプログラムです。

このプログラムは、シミュレーションの実行に必要な各種プログラムの自動実行や、プログラム間の同期を取るといった処理を行います。

また、シミュレーションを行うためのシステム・コンフィギュレーションの設定、使用するコンフィギュレーションや各種データを指定するためのプロジェクトの設定を行うことが出来ます。

Score以外の構成要素としては、以下の4つのプログラムがあります。

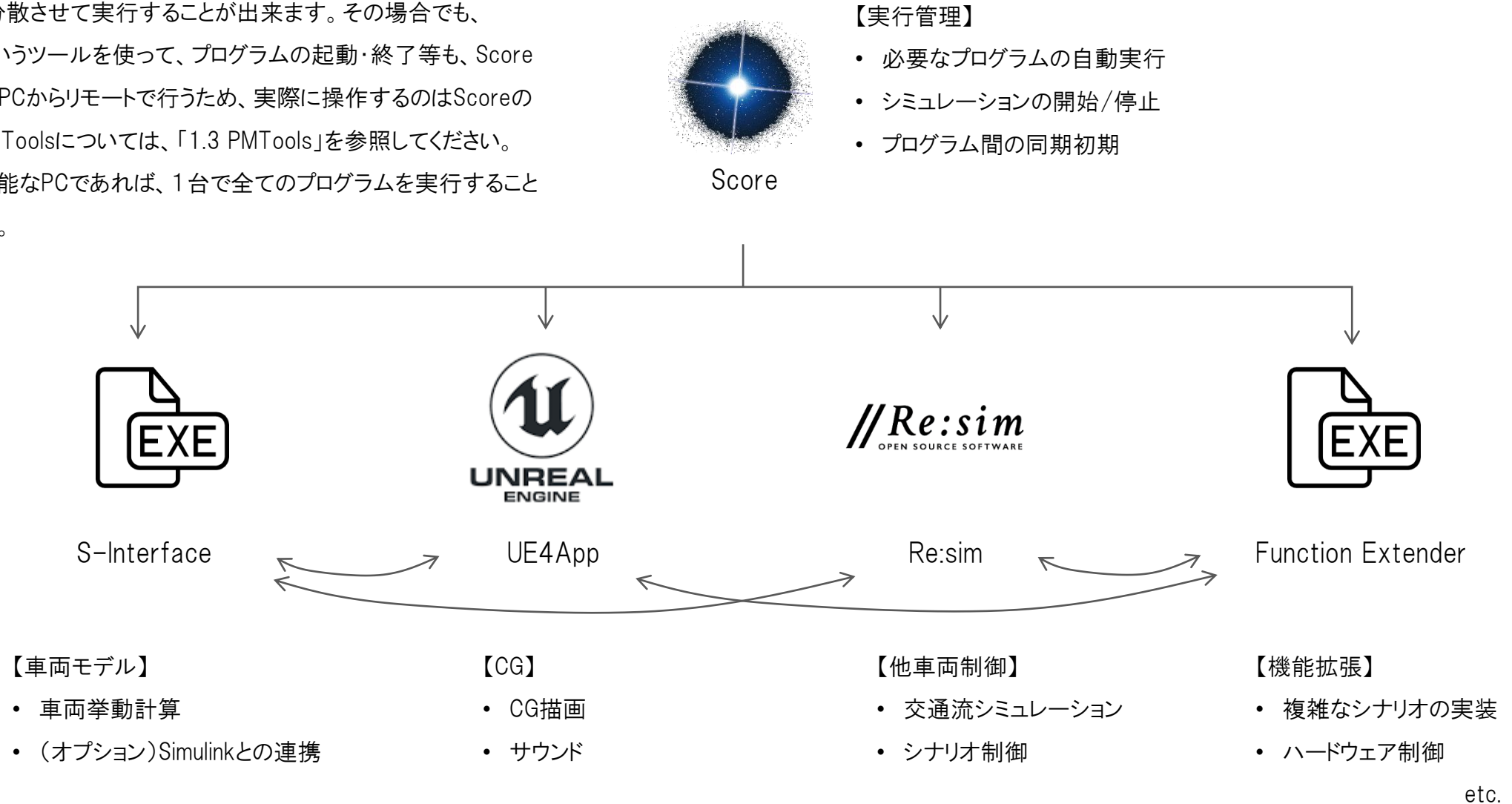
S-Interface(エス・インターフェース)	: 車両モデルを実装したプログラム
Re:sim(リシム)	: 他車両制御・シナリオ実現のためのプログラム
UE4 App(ユーイーフォー・アップ)	: UE4のパッケージ化プログラム(CG描画プログラム)
Function Extender(FE・エフ・イー)	: 機能拡張のためのユーザー定義プログラム

* Re:simとUE4は、オープンソースのプログラムですので、難易度は非常に高いですが、独自にカスタマイズすることは可能です。

ドライビングシミュレータの機能を実現するために、最低限必要なプログラムは、S-InterfaceとUE4Appです。

他車両や歩行者・自転車などを登場させないのであればRe:simは不要ですし、簡単なシナリオを実現するだけであればFEは不要です。

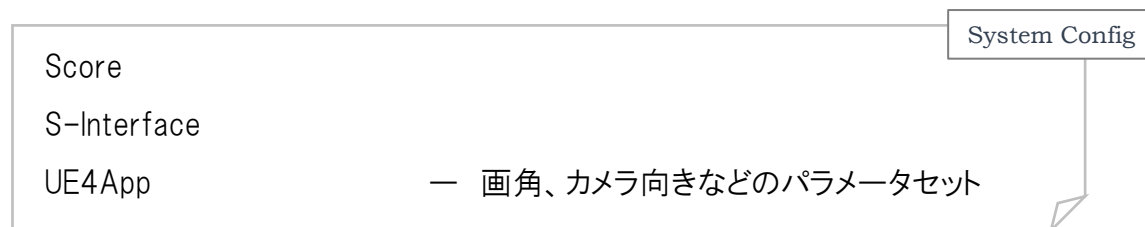
プログラム間は、ネットワーク通信によりデータを授受するので、複数台のPCに分散させて実行することが出来ます。その場合でも、PMToolsというツールを使って、プログラムの起動・終了等も、Scoreを実行するPCからリモートで行うため、実際に操作するのはScoreのみです。PMToolsについては、「1.3 PMTools」を参照してください。なお、高性能なPCであれば、1台で全てのプログラムを実行することも可能です。



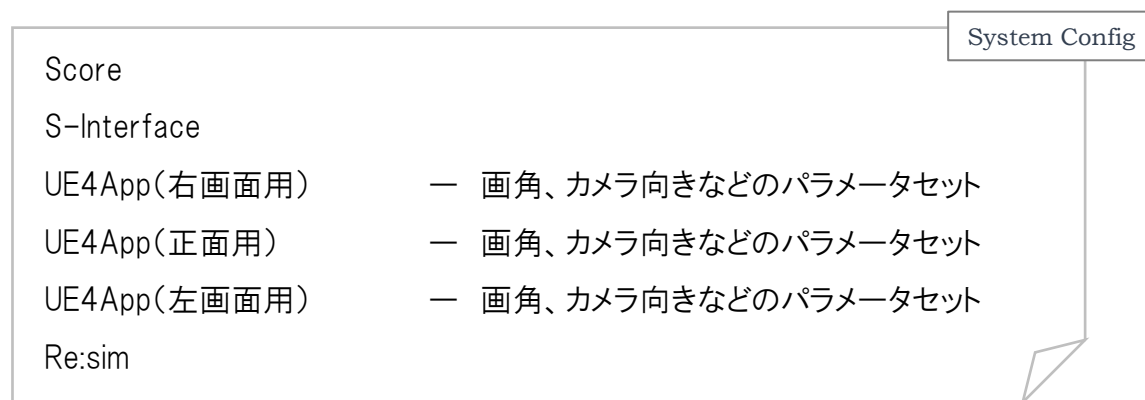
1.1 システム・コンフィギュレーション

システム・コンフィギュレーションは、シミュレーションの実行を行うために、どのプログラムを、どのように使用するか、をまとめたデータセットです。

たとえば、1画面のみ、DS車両のみで、他車両などのオブジェクトは出現しないような場合のシステム・コンフィギュレーションは、



となりますし、3台のPCで3画面描画して、他車両を交通流シミュレーションで出現させるような場合のシステム・コンフィギュレーションは、



となります。

UE4Appは、UE4のプロジェクトをパッケージ化して、単独に実行できるようにした実行ファイルのニックネームとして使用しており、実際には、UE4のプロジェクト名に応じたものとなります。

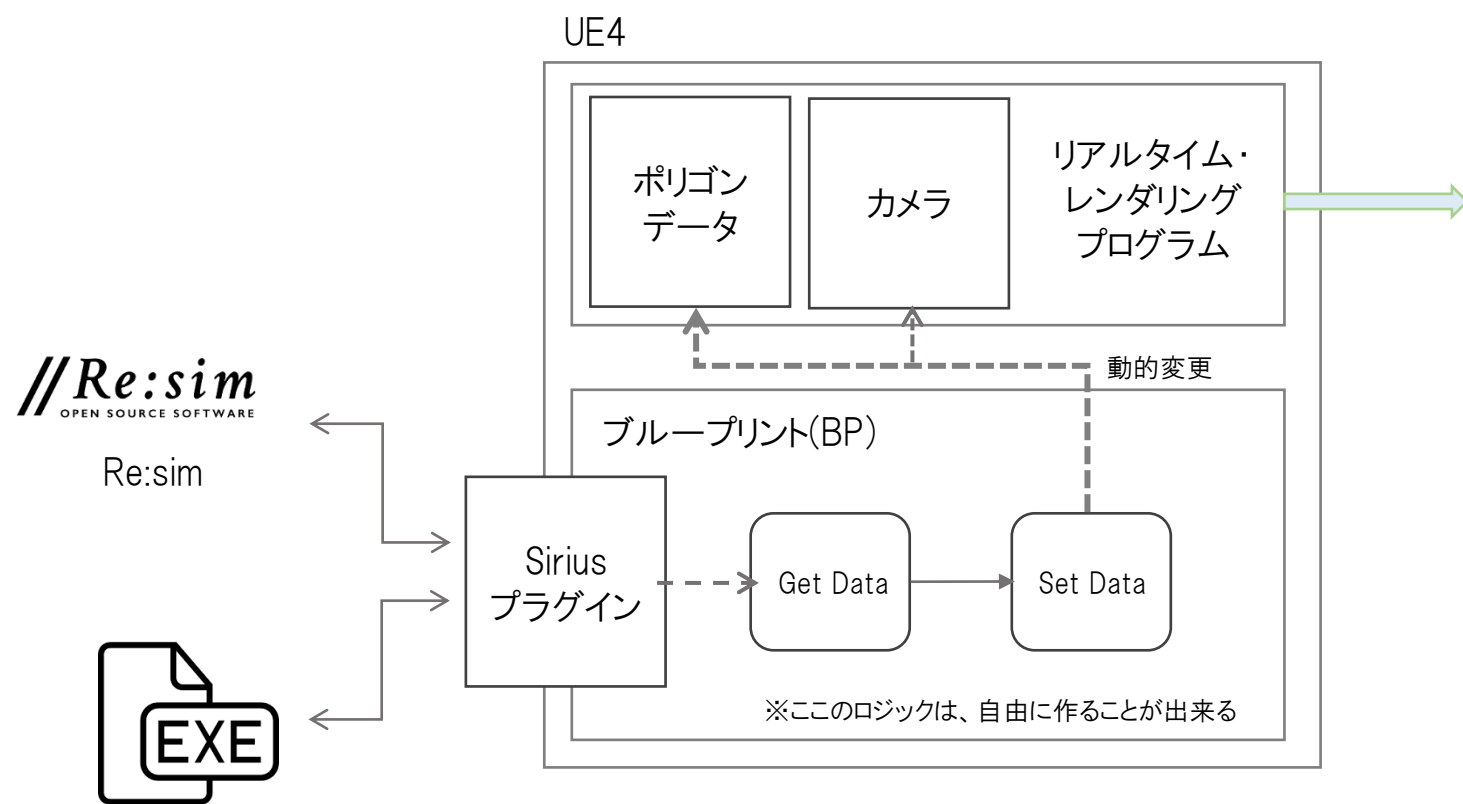
UE4の1つのプロジェクトは、1つのマップに対応し、また同じマップでも、画面に警報等を表示する機能を持たせるか否か、などによって、異なるプロジェクトを用意することになります。

マップ	機能	UE4プロジェクト	パッケージ化
マップA	CG表示のみ	Map_A_Base	→ Map_A_Base.exe
	FEからデータを受け取って 警報等表示	Map_A_Assist	→ Map_A_Assist.exe
マップB	CG表示のみ	Map_B_Base	→ Map_B_Base.exe
	FEからデータを受け取って 太陽位置変更(朝→昼→ 夜の変化)	Map_B_TimeVarying	→ Map_B_TimeVarying.exe

※表中の名称等はサンプルであり、実際のデータではありません。

システム・コンフィギュレーションにおいて設定するUE4Appは、上の表示におけるUE4プロジェクトをパッケージ化して出来るプログラムのことです。

UE4は、ゲームエンジンと呼ばれるもので、単にCGを描画するだけの機能ではなく、ユーザーが独自にプログラムを作成して、ロジックを追加することが可能で、大変高機能なソフトウェアです。Siriusでは、三咲デザインが開発したSiriusプラグインを使用することで、Re:simやFEなどからデータを受信して利用できるようにしています。そのデータをどのように利用するかは、UE4のブループリントと呼ばれるビジュアルスクリプティング機能でプログラムを作成します。

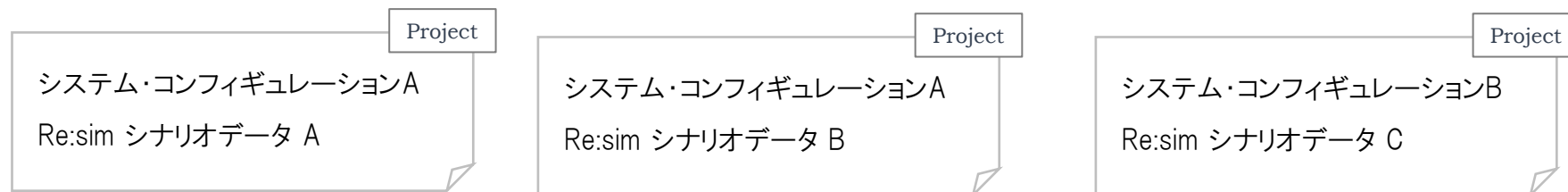


※UE4は、オープンソースのソフトウェアであり、Siriusはそれを利用しているだけであり、UE4の使い方については別途、書籍やインターネットなどで調べて頂く必要があります(有償になりますが、UE4教育のご要望がありましたら検討いたします)。

1.2 プロジェクト

Siriusにおけるプロジェクトは、シミュレーションを実行するための、システム・コンフィギュレーションおよびシナリオデータをまとめたデータセットです。

同じマップ(システム・コンフィギュレーション)でも、使用するシナリオデータが異なれば、別のプロジェクトとして設定します。



Re:simシナリオデータは、SEdit(エス・エディット)を使って作成します。





SEditについては、SEdit-Manual-Part1-Basic_Operation.pdf と SEdit-Manual-Part2-Scenario_Data_Setting.pdf を参照してください。

1.3 PMTools

Siriusでは、PMTools内のPsExecというプログラムを使用して、リモートPCでのプログラムの自動実行を行います。
Cドライブ直下に、PMToolsをコピーしておく必要があります。

<https://docs.microsoft.com/en-us/sysinternals/downloads/psexec>

PsExec v2.2

06/29/2016 • 4 minutes to read •    

By Mark Russinovich

Published: June 29, 2016



[Download PsTools](#) (2.7 MB)

Introduction

Utilities like Telnet and remote control programs like Symantec's PC Anywhere let you execute programs on remote systems, but they can be a pain to set up and require that you install client software on the remote systems that you wish to access. PsExec is a light-weight telnet-replacement that lets you execute processes on other systems, complete with full interactivity for console applications, without having to manually install client software. PsExec's most powerful uses include launching interactive command-prompts on remote systems and remote-enabling tools like IpConfig that otherwise do not have the ability to show information about remote systems.

Note: some anti-virus scanners report that one or more of the tools are infected with a "remote admin" virus. None of the PsTools contain viruses, but they have been used by viruses, which is why they trigger virus notifications.

Installation

Just copy PsExec onto your executable path. Typing "psexec" displays its usage syntax.

Using PsExec

See the July 2004 issue of *Windows IT Pro Magazine* for [Mark's article](#) that covers advanced usage of PsExec.

Usage: psexec [\\computer[,computer2[,...]] | @file\]][-u user [-p psswd]][-n s][[-r servicename][[-h][[-l][[-s][-e][[-x][[-i [session]]][[-c executable [-f[-v]]][-w directory][[-d][[-<priority>]][-a n,n,...] cmd [arguments]

PMToolsを使えるようにするための設定

①ポートを通す(FirewallをOFFにしているときは、おそらく不要)

```
135
455
%SystemRoot%\system32\services.exe
```

Windowsのセキュリティ設定で、Access Deniedとなることが多く、多少面倒です。

これまでの経験上、Firewall, UACを無効にした上で、レジストリの修正とグループポリシーの変更を行えば、大体は使えるようになります。

②UAC(User Access Control)を通知しないにする

③レジストリの修正

```
reg add "HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System" /v "LocalAccountTokenFilterPolicy" /t REG_DWORD /d 1
```

④パスワードなしでログインできるようにする場合

gpedit.mscをインストール(*)

gpedit.msc(ローカルグループポリシーエディタ)を実行して、「カウント:ローカルアカウントの空のパスワードの使用をコンソールログオンのみに制限する」を無効にする

(*) 以下をバッチファイルで実行する

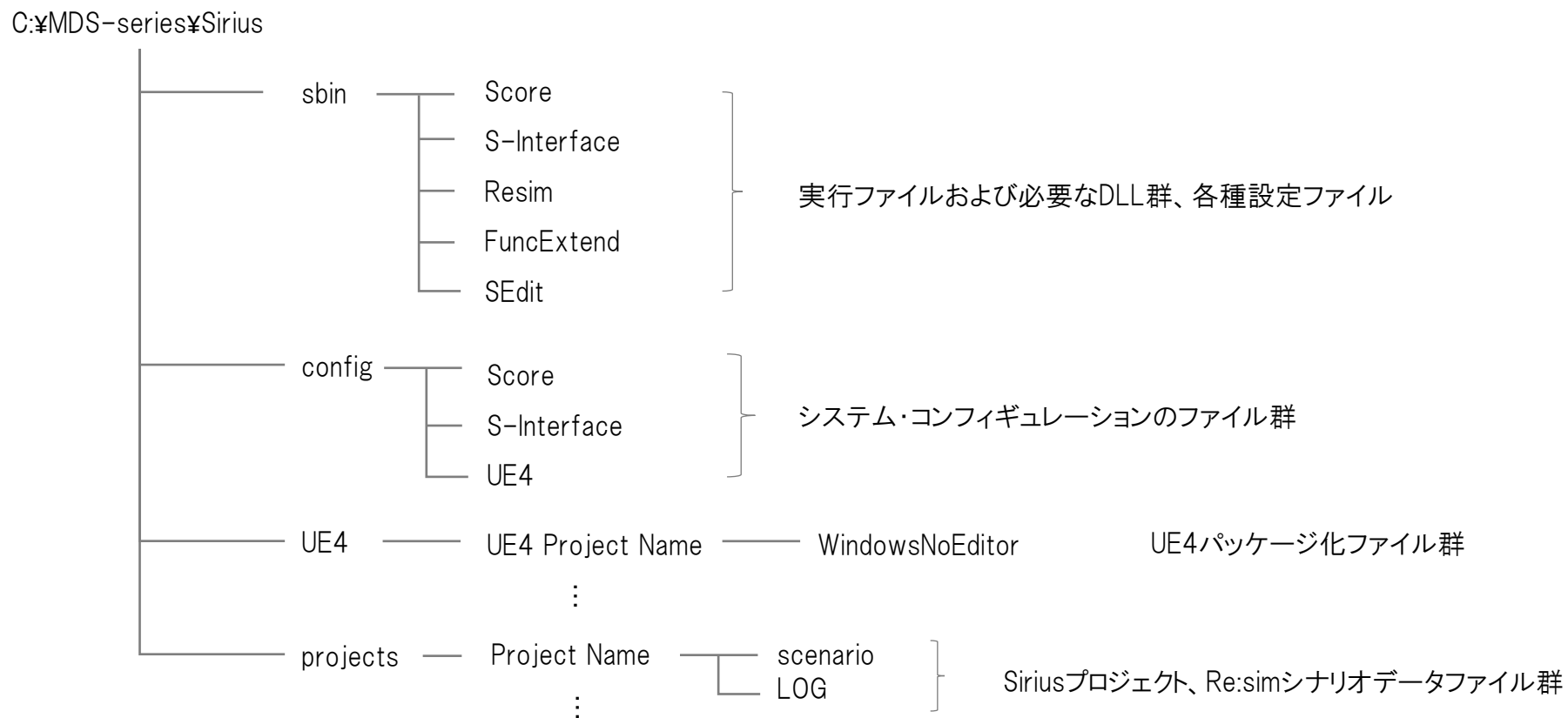
```
@echo off
pushd "%~dp0"

dir /b %SystemRoot%\servicing\Packages\Microsoft-Windows-GroupPolicy-ClientExtensions-Package~3*.mum >List.txt
dir /b %SystemRoot%\servicing\Packages\Microsoft-Windows-GroupPolicy-ClientTools-Package~3*.mum >>List.txt

for /f %i in ('findstr /i . List.txt 2^>nul') do dism /online /norestart /add-package:"%SystemRoot%\servicing\Packages\%i"
pause
```

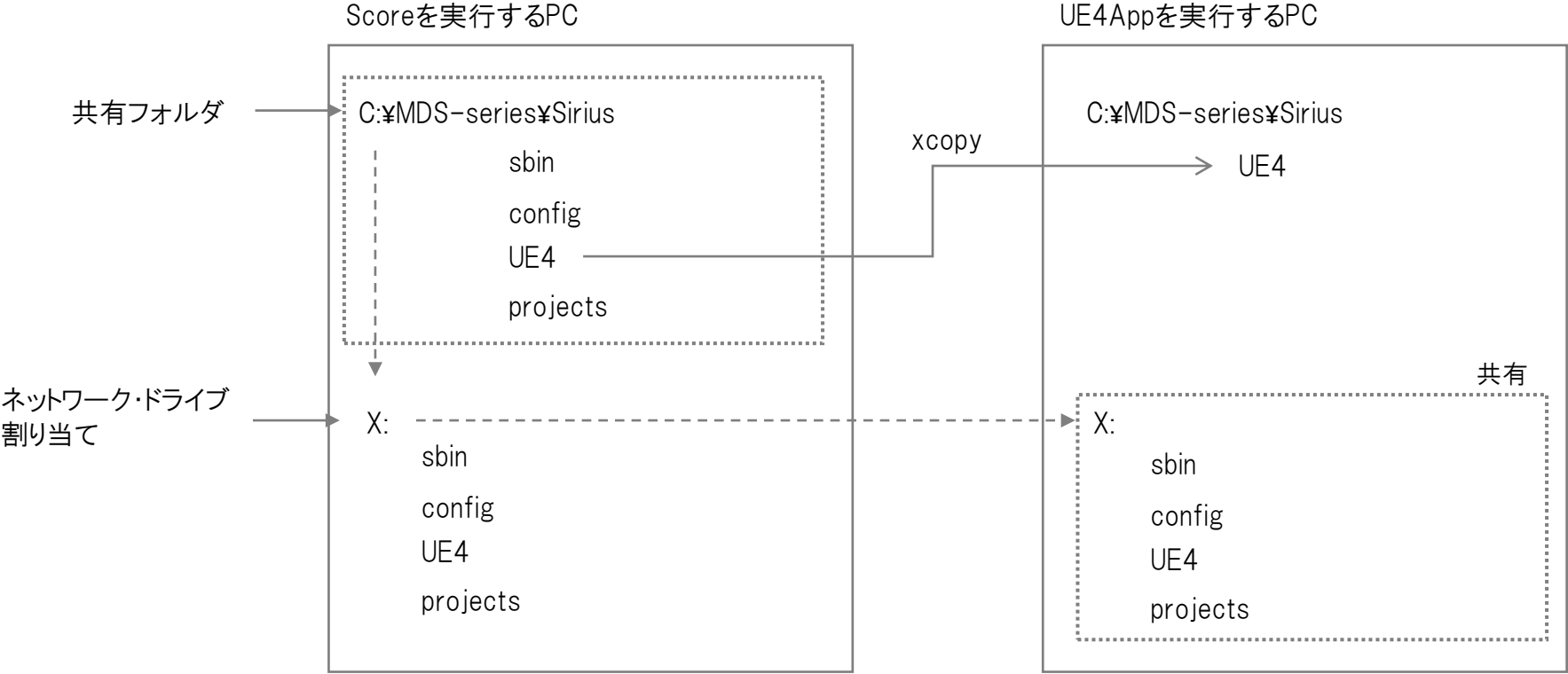
2. フォルダ構成

Siriusは、以下のフォルダ構成を前提として作成されています。



Siriusプロジェクトについては、必ずしも上記のフォルダ構成に従う必要はありませんが、バックアップを取るなどを考えると、色々なところにフォルダを作るよりは、上記projectsフォルダ内にサブフォルダを作り、格納した方が手間が掛かりません。

複数台のPCに分散させて実行させる場合、Scoreを実行するPCのC:¥MDS_series¥Siriusフォルダを共有フォルダとして、他のPCから見れるようにしておく必要があります。また、UE4を実行するグラフィックPCについては、UE4のパッケージ化ファイル群をローカル環境に予めコピーしておく必要があります。その手順については、本Technical Manualの「5. UE4との連携」をご覧ください。



3. Score

3.1 GUI

ネットワーク情報設定

プリファレンス設定

システム・コンフィギュレーション
設定

プロジェクト設定

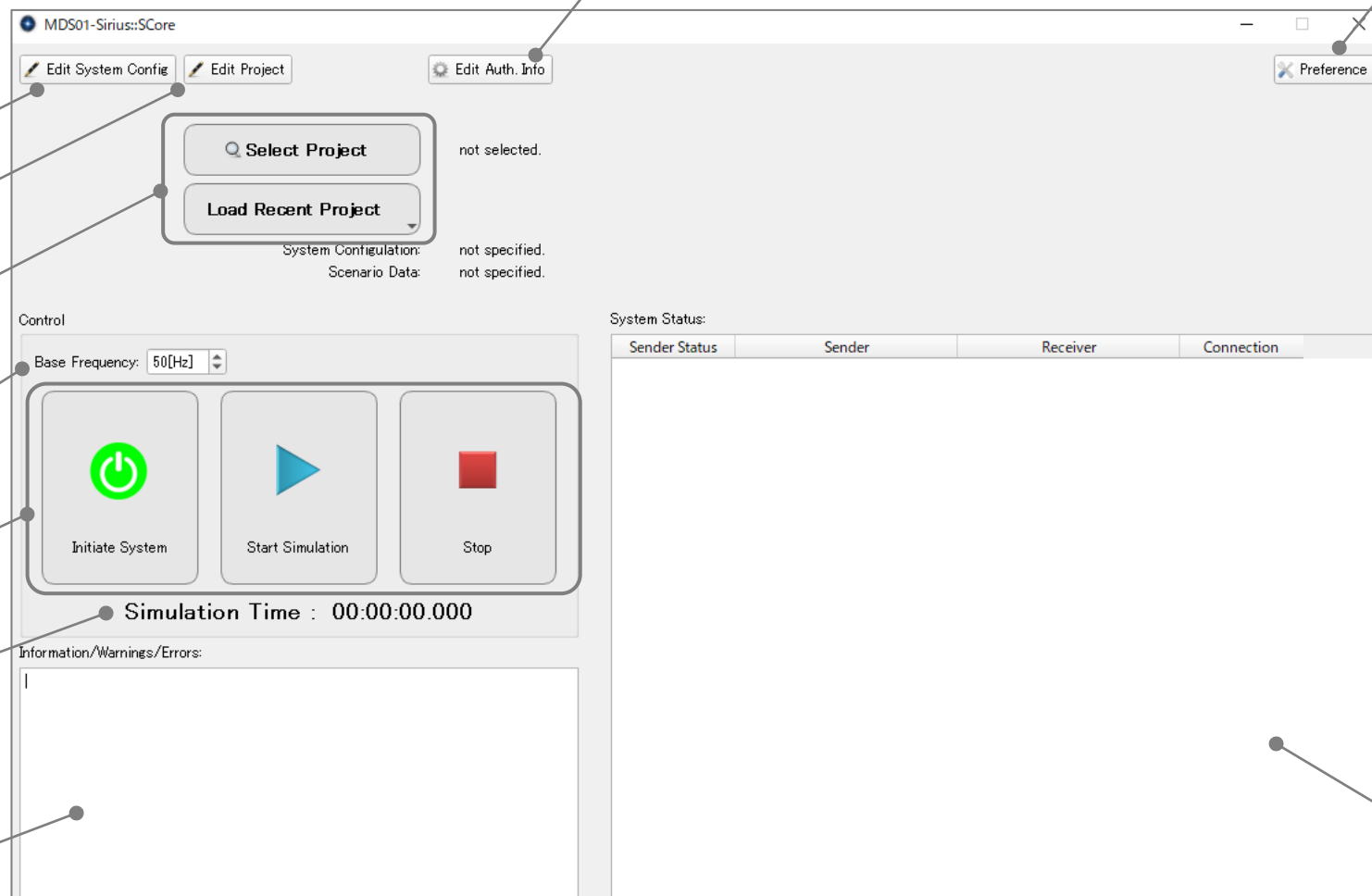
プロジェクト選択

実行周期設定

シミュレーション操作ボタン

シミュレーション時刻表示

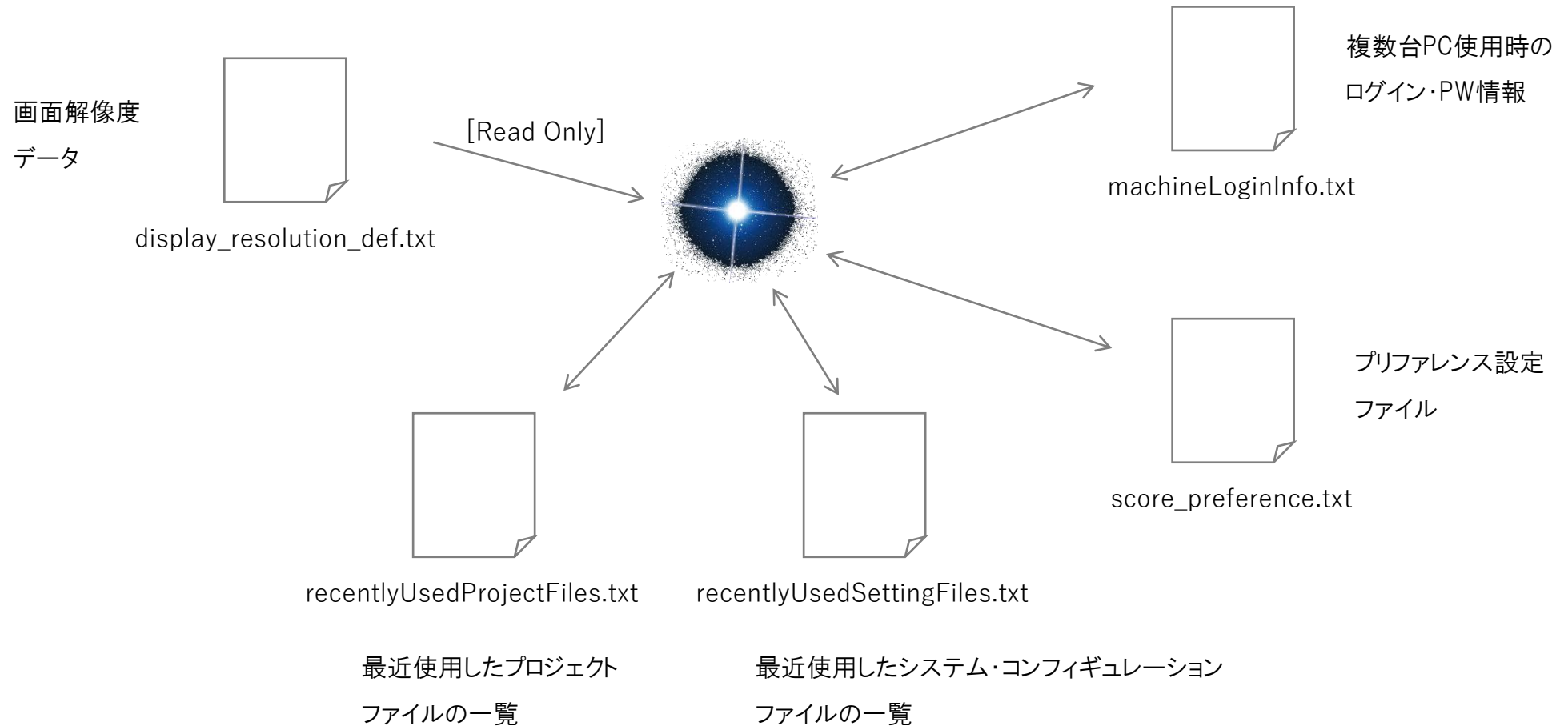
メッセージ表示部



通信状態表示部

3.2 Score関連設定ファイル

sbin/Scoreに置かれており、Scoreが実行時に読み込む、もしくは、実行中にロード/セーブするファイルとして、以下のがあります。
display_resolution_def.txtのみ、手動で設定し、ほかはScoreのGUIにて編集することができます。



3.3 システム・コンフィギュレーションの編集

“Edit System Config”ボタンを押すと、以下のダイアログが表示されます。

コンフィギュレーション・ファイルのロード、セーブは普通のアプリケーションと同じように行います。

MDS01-Sirius::SCore System Configuration Setting - No Config loaded.

Load Recent

Node Definition: Add Node Edit Node Del Node Up Down

Node	Executable	Options	IP-Address
1 SCore	SCore.exe		192.168.1.3

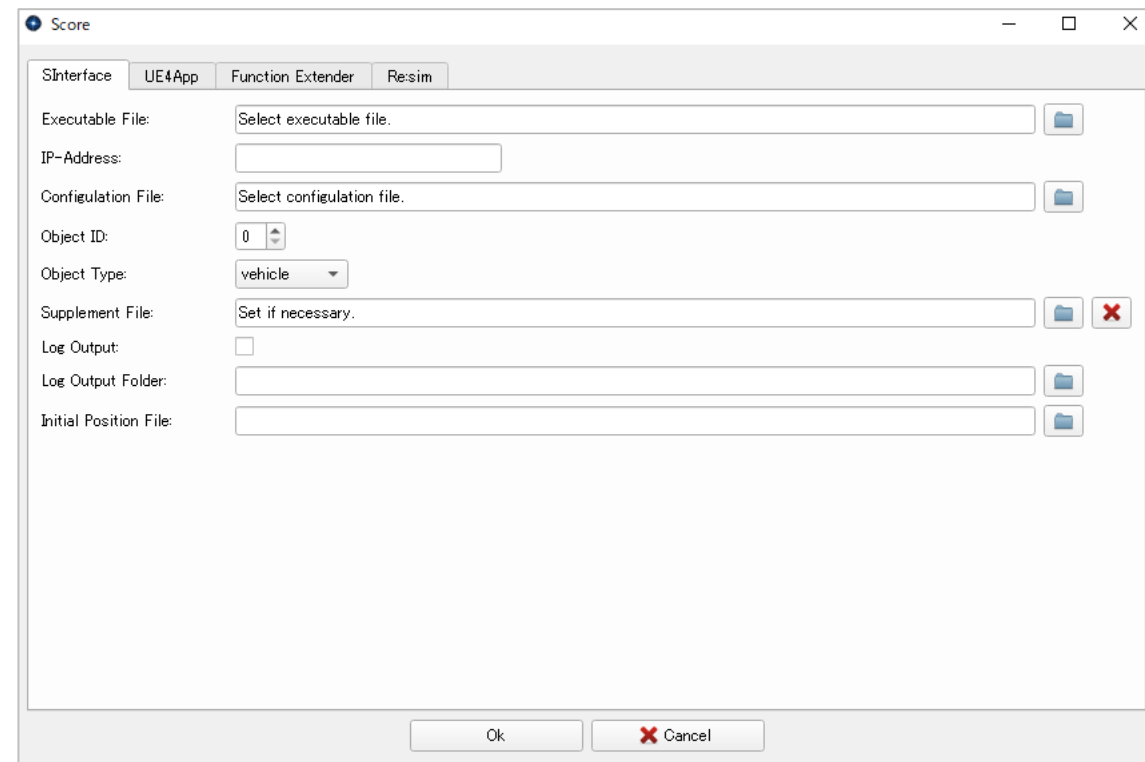
Port Setting:

Sender	Receiver	Port
--------	----------	------

Configuration Check:

”Add Node”ボタンを押して、使用するプログラムを追加・設定します。

以下のようなダイアログが表示されますので、必要項目を設定していきます。



The image shows a Windows-style dialog box titled "Score". It has four tabs: "SInterface", "UE4App" (which is selected), "Function Extender", and "Resim". The "UE4App" tab contains the following fields and controls:

- Executable File:** A text field with the placeholder "Select executable file." and a folder selection icon on the right.
- IP-Address:** A text field.
- Configuration File:** A text field with the placeholder "Select configuration file." and a folder selection icon on the right.
- Object ID:** A numeric input field with the value "0" and up/down arrow buttons.
- Object Type:** A dropdown menu currently showing "vehicle".
- Supplement File:** A text field with the placeholder "Set if necessary.", a folder selection icon, and a red "X" icon.
- Log Output:** A checkbox that is currently unchecked.
- Log Output Folder:** A text field with a folder selection icon on the right.
- Initial Position File:** A text field with a folder selection icon on the right.

At the bottom of the dialog, there are two buttons: "Ok" and "Cancel" (with a red "X" icon).

① S-Interface

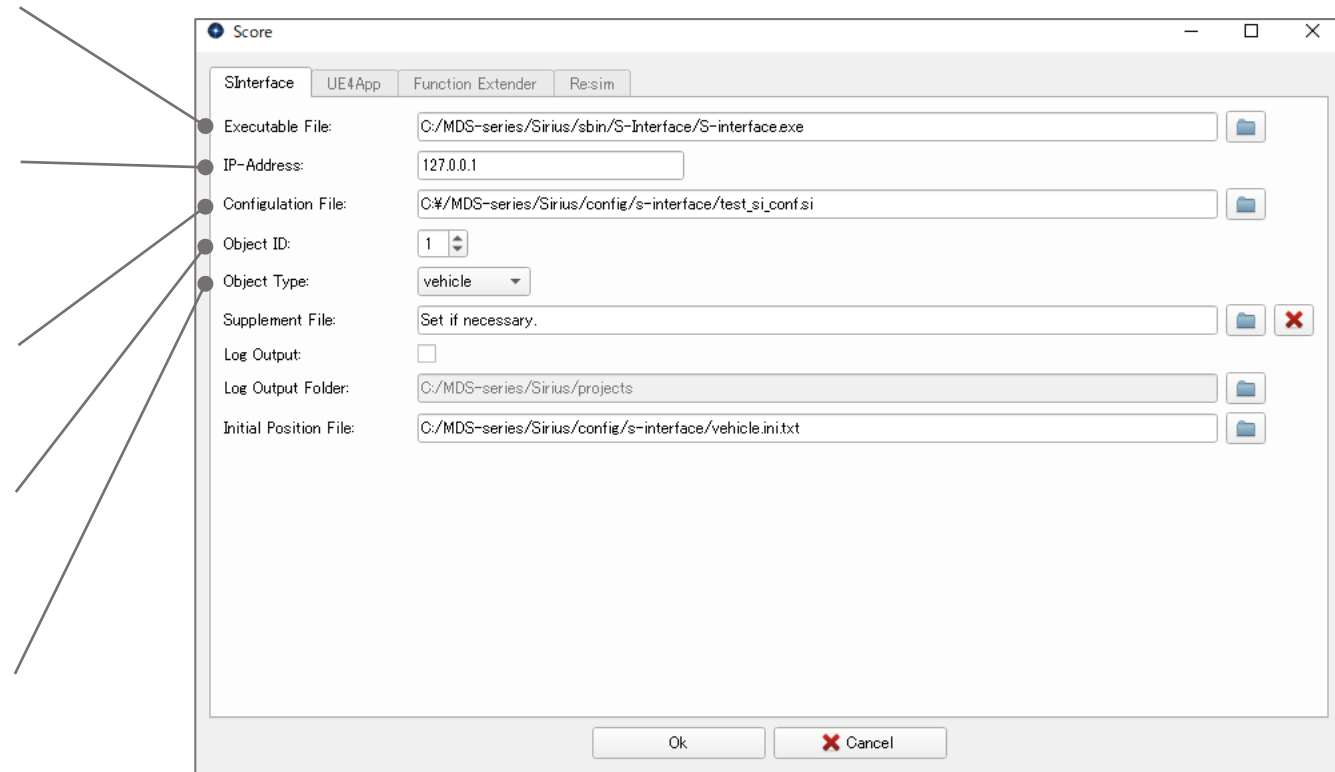
使用するS-Interfaceの実行ファイルを選択します
(実行ファイルは通常はsbin/S-Interfaceに格納する)

S-Interfaceを実行するPCのIPアドレスを指定します。
PC1 台のみの場合は127.0.0.1か、そのPCのIPアドレス
を入力します。

S-Interfaceのコンフィギュレーション・ファイル名を指定
します。

S-InterfaceオブジェクトのID番号を指定します。Re:sim,
UE4, FEでは、このID番号でS-Interfaceオブジェクトを
識別します。

S-Interfaceオブジェクトの種類を指定します。
vehicle / pedestrian の2種類です。



S-Interfaceで必要とするファイルを設定できます。
Simulink利用時は、ここにSimulinkモデル名を指定します。

S-Interfaceでログ出力するかどうかの設定です。
S-Interfaceのプログラムで対応している必要があります。

S-Interfaceでログ出力する場合の出力先の指定です。

S-Interfaceオブジェクトの初期位置を設定している
テキストファイルを設定できます。
テキストファイルのフォーマットは、

X[m] ; -100.0

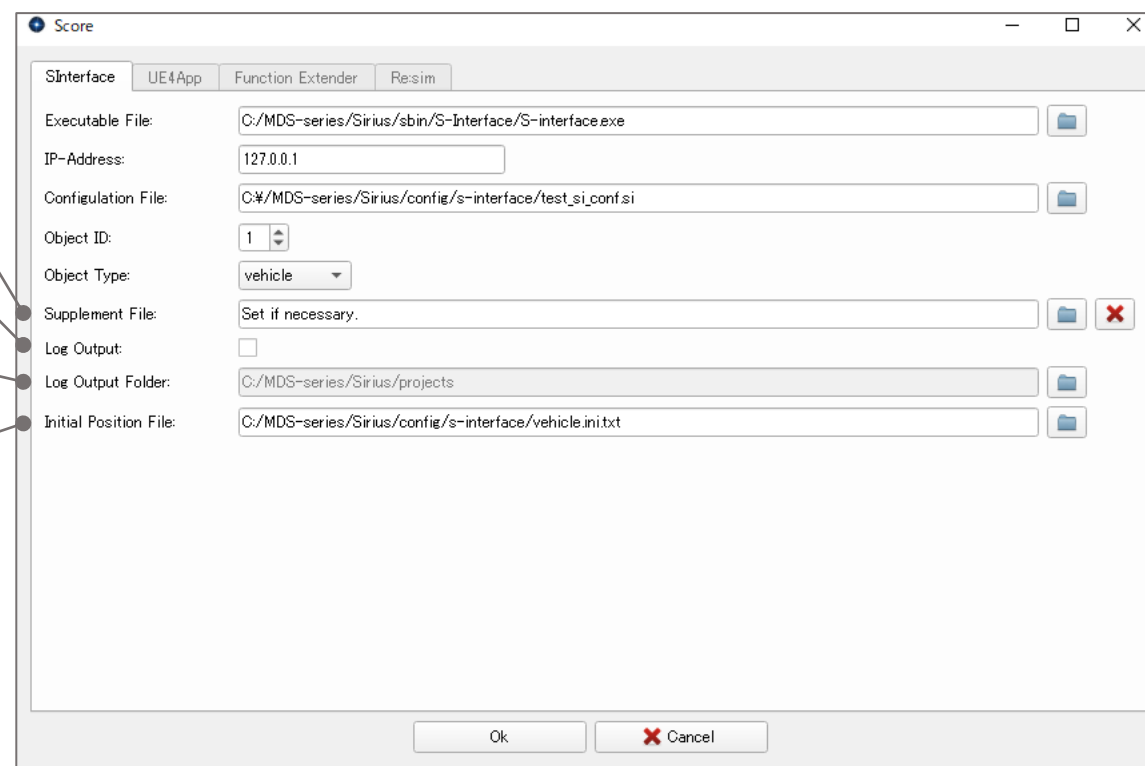
Y[m] ; 0.0

Z[m] ; 0.0

Yaw[deg] ; 0

V[km/h] ; 0

のように、5つの項目をUE4の座標系で設定します。



② UE4App

UE4Appを使用する場合は、以下を設定します。

パッケージ化したUE4プロジェクトの実行ファイルを指定します。通常、WindowsNoEditor直下にあります。

UE4Appを実行するPCのIPアドレスを指定します。
PC1台のみの場合は127.0.0.1か、そのPCのIPアドレスを入力します。

UE4Appのコンフィギュレーション・ファイル名を指定します。UE4Appは、このファイルを読み込んで、以下で設定する情報を取得します。

S-Interfaceオブジェクトの動きに合わせてCG描画を行うために、S-Interfaceで設定するオブジェクトIDを設定します。

複数のPCでCGを表示させるためのスクリーンのID番号を設定します。0からの連番にします。

Score

Interface UE4App Function Extender Resim

Executable File: C:/MDS-series/Sirius/UE4/UE4Project/WindowsNoEditor 1/UE4Project.exe

IP-Address: 127.0.0.1

Configuration File: C:/MDS-series/Sirius/config/UE4/test_ue4_setting_s0ue4conf

Associated Object ID: 1

Screen ID: 0

Log Output: ☐ Log Output

Max Actor: 60

Display Resolution: 800x600

Full Screen: ☐ Full Screen

Camera Direction[deg]: 0

Camera Field of View[deg]: 90

Camera Aspect Ratio[-]: 1.77

Camera Position Offset[m]: X: 0.11 Y: 0.3 Z: 1.10

Supplement File: Set if necessary.

Ok Cancel

ログファイルを出力するかどうかの設定です。デバッグ用途なので、通常はチェックを入れません。

UE4で扱うアクターの最大数を設定します。現在のところ、車両、歩行者、自転車などを合わせて、1000を設定しておきます。

表示するCGの解像度を指定します(次のフルスクリーン設定が無効な場合のみ)。

フルスクリーン表示にするかどうかを設定します。

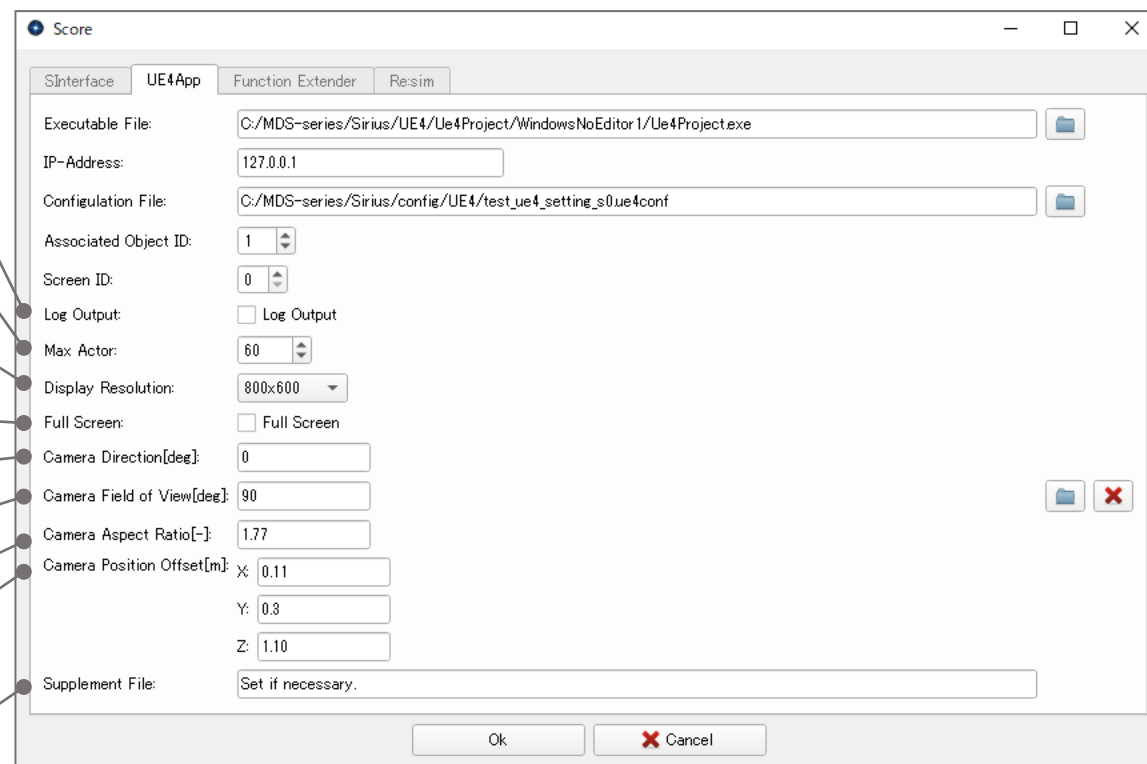
カメラの正面方向を指定します。

カメラの画角を指定します。

カメラのアスペクト比を指定します。

S-Interfaceオブジェクトの座標中心からカメラ取付位置までのオフセット量を指定します。

UE4Appが別途読み込むファイルを指定できますが、現在はSiriusプラグイン側が対応していません。



③ Function Extender

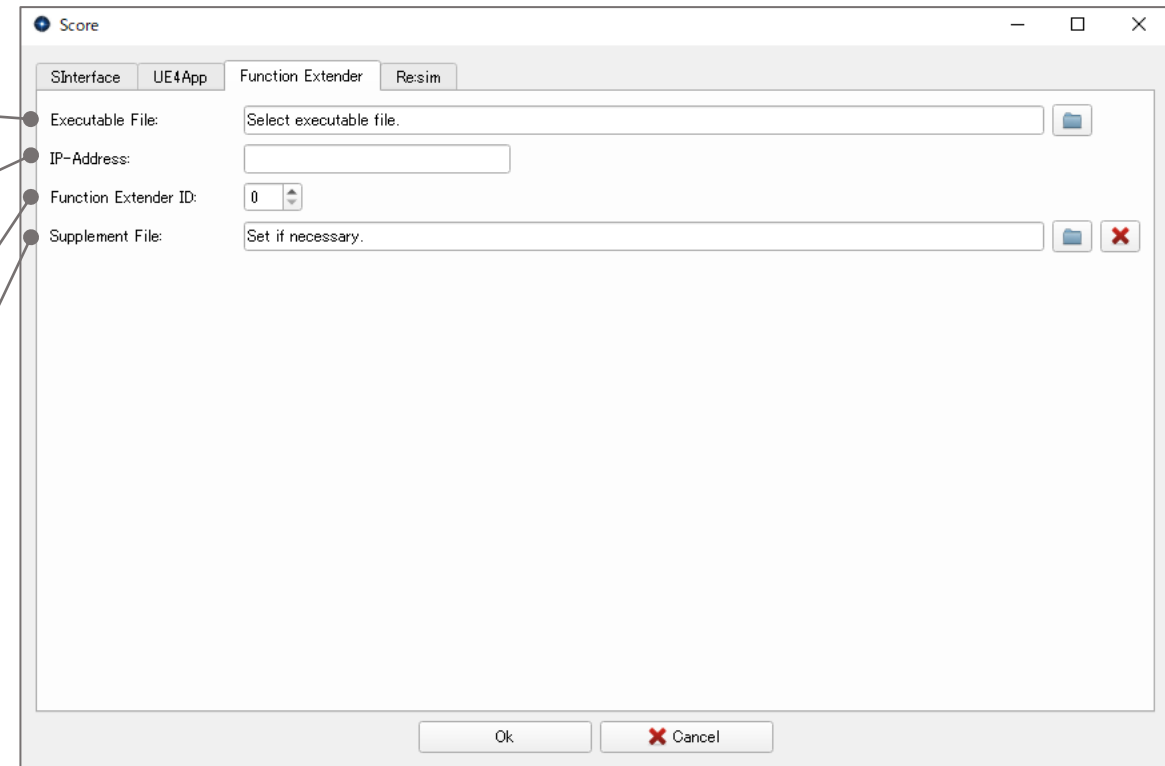
FEを使用する場合は、以下を設定します。

FEの実行ファイルを選択します(実行ファイルは通常は
sbin/FuncExtendに格納する)

FEを実行するPCのIPアドレスを指定します。
PC1台のみの場合は127.0.0.1か、そのPCのIPアドレス
を入力します。

FEを複数実行する場合のために、ID番号を設定します。
1つだけの場合はデフォルト値(0)のままにします。

FEの実行時に読み込むFE用設定ファイルを指定できま
す。



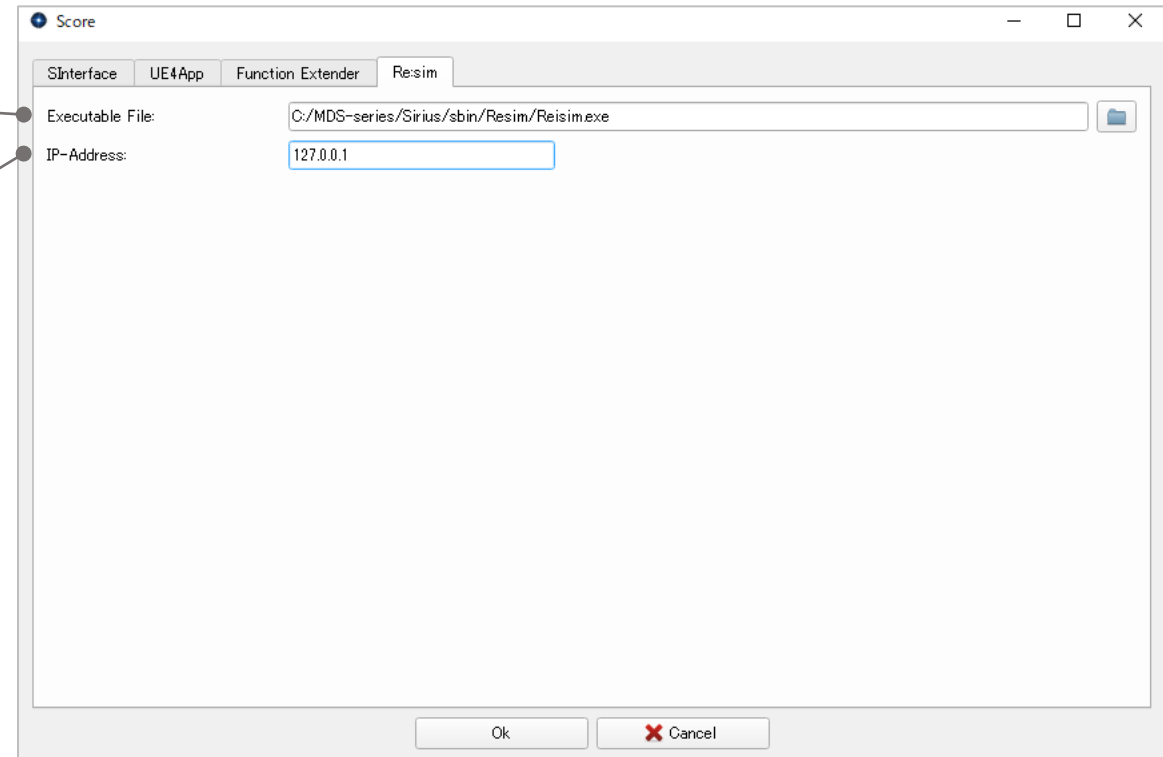
④ Re:sim

Re:simを使用する場合は、以下を設定します。

sbin/Resim/Reisim.exeを指定します。

PC1台のみの場合は C:\MDS-series/Sirius,
複数台PCの場合は X:です。

Re:simを実行するPCのIPアドレスを指定します。
PC1台のみの場合は127.0.0.1か、そのPCのIP
アドレスを入力します。



設定内容を編集するためには、変更したいNodeのExecutableのセルをクリックしてから、“Edit Node”ボタンを押します。

削除する場合にも、削除するNodeのExecutableのセルをクリックしてから、“Del Node”ボタンを押します。

上下矢印ボタンで、表中の順序を変更することが出来ますが、見た目だけの問題で、実行において順序は関係ありません。

MDS01-Sirius:SCore System Configuration Setting - C:/MDS-series/Sirius/config/score/01Highway.sconf

Load Recent

Node Definition:

Add Node

Edit Node

Del Node

Node	Executable	Options	IP-Address
1 SCore	SCore.exe		192.168.1.3
2 S-Interface	C:/MDS-series/Sirius/sbin/S-Interface/S-interface.exe	<div>-Conf=C:/MDS-series/Sirius/config/s-interface/test_si_conf.si -id=1 -type=vehicle -Log=0 -Dir=C:/MDS-series/Sirius/projects/Shizuoka/Log -Init=C:/MDS-series/Sirius/config/s-interface/vehicle.ini.txt</div>	127.0.0.1
3 UE4App	C:/MDS-series/Sirius/UE4/UE4Project/WindowsNoEditor1/UE4Project.exe	<div>-Conf=C:/MDS-series/Sirius/config/UE4/test_ue4_setting_s0_ue4conf -screen=0 -Log=0 -MaxActor=60 -resolv=800x600w -direct=0 -FoV=90 -AR=1.77 -Pos=(0.11,0.3,1.10) -object=1</div>	127.0.0.1
4 Resim	C:/MDS-series/Sirius/sbin/Resim/Resim.exe		127.0.0.1

Port Setting:

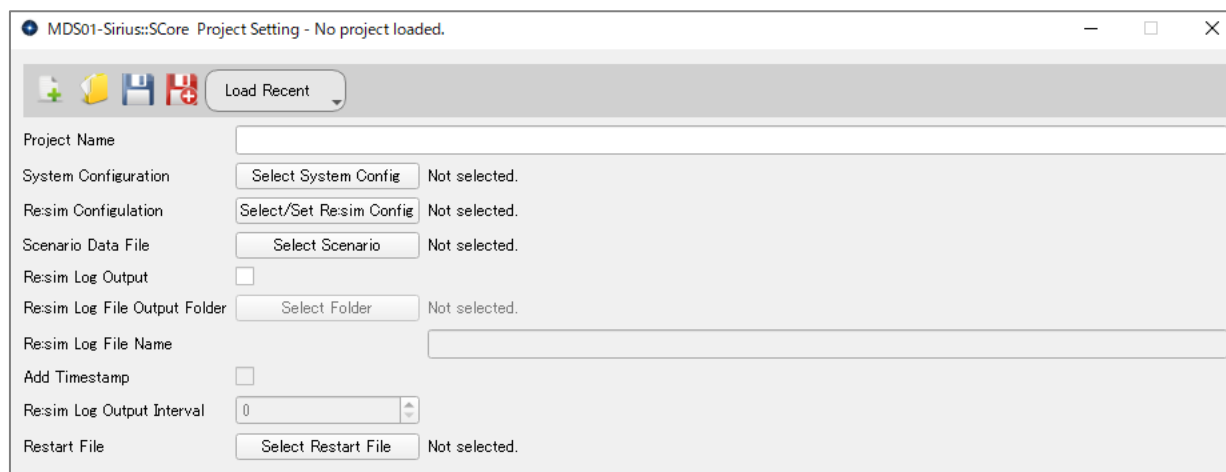
	Sender	Receiver	Port
1	SCore	S-Interface[1]	58000
2	SCore	UE4App[1,0]	58001
3	SCore	Resim	58002
4	S-Interface[1]	SCore	58100
5	S-Interface[1]	UE4App[1,0]	58101
6	S-Interface[1]	Resim	58102
7	UE4App[1,0]	SCore	58200
8	UE4App[1,0]	S-Interface[1]	58201
9	UE4App[1,0]	Resim	58202

Configuration Check:

3.4 プロジェクトの編集

“Edit Project”ボタンを押すと、以下のダイアログが表示されます。

プロジェクト・ファイルのロード、セーブは普通のアプリケーションと同じように行います。



プロジェクトの名称を付けることが出来ます。この名称が”Load Recent Projects”に表示されます。

使用するシステム・コンフィギュレーションを選択します。事前に設定しておく必要があります。

Re:simを動かすためのRe:simコンフィギュレーションのファイル名を指定します。

SEditで作成したRe:simシナリオファイルを指定します。

Re:simでログファイルを出力するかどうかを指定します。

Re:simでログ出力する際のログファイル出力先を指定します。

Re:simでログ出力する際のログファイル名を指定します。

ログファイル名に、タイムスタンプを付けるかどうかを指定します。

ログデータ出力の間隔を指定します。出力間隔は
(ここで設定した間隔+1)/実行周期 [s] となります。

Re:simのRestart機能を使う場合に、Restartファイルを指定します。

MDS01-Sirius::SCore Project Setting - No project loaded.

Project Name

System Configuration Select System Config Not selected.

Re:sim Configuration Select/Set Re:sim Config Not selected.

Scenario Data File Select Scenario Not selected.

Re:sim Log Output ☐

Re:sim Log File Output Folder Select Folder Not selected.

Re:sim Log File Name

Add Timestamp ☐

Re:sim Log Output Interval 0

Restart File Select Restart File Not selected.

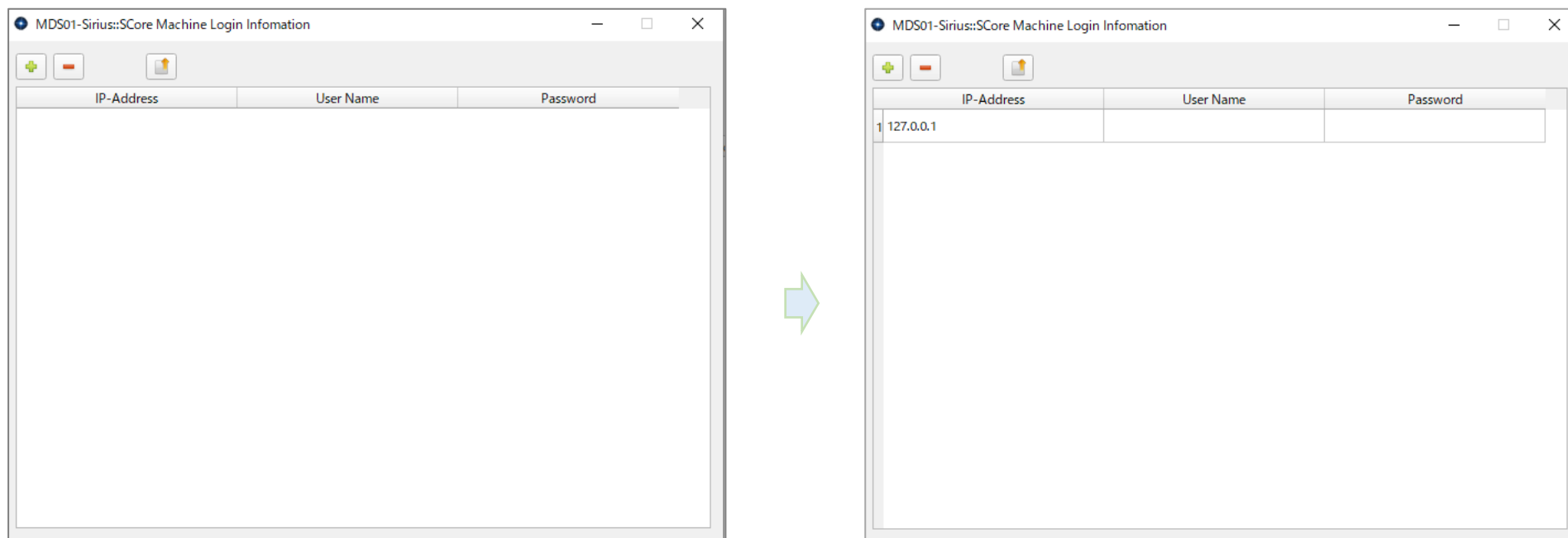
3.5 他PCへのログイン情報の編集

“Edit Auth. Info”ボタンを押すと、以下のダイアログが表示されます。

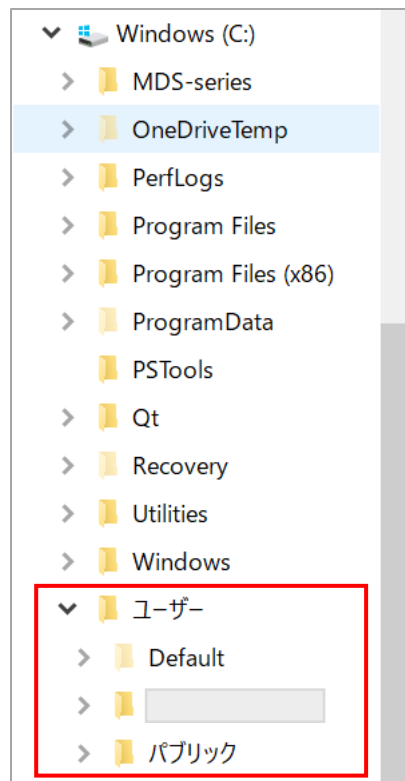
“+”ボタンを押すと、1行追加されて、使用するPCのIPアドレスとログインするためのユーザー名、パスワードを入力することが出来ます。

このダイアログを閉じると、入力内容は machineLoginInfo.txt に自動的に保存されます。

一台で動かすときは、特に設定しておく必要はありません。



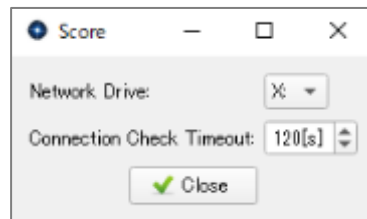
ユーザー名は、エクスプローラーで表示されるC:\Users以下にある名前でない、ログインできませんので、ご注意ください。
(最初に登録したユーザー名とWindowsで表示されるユーザー名が変更されている場合があります)



パスワードを設定していない場合は、何も入力しないでください。

3.6 プリファレンス設定

“Preference”ボタンを押すと、以下のダイアログが表示されます。



現状では、ネットワーク・ドライブと、通信チェック時のタイムアウト時間を設定することが出来ます。

デフォルト値は、ネットワーク・ドライブが “X:”、タイムアウトが 120[s] です。

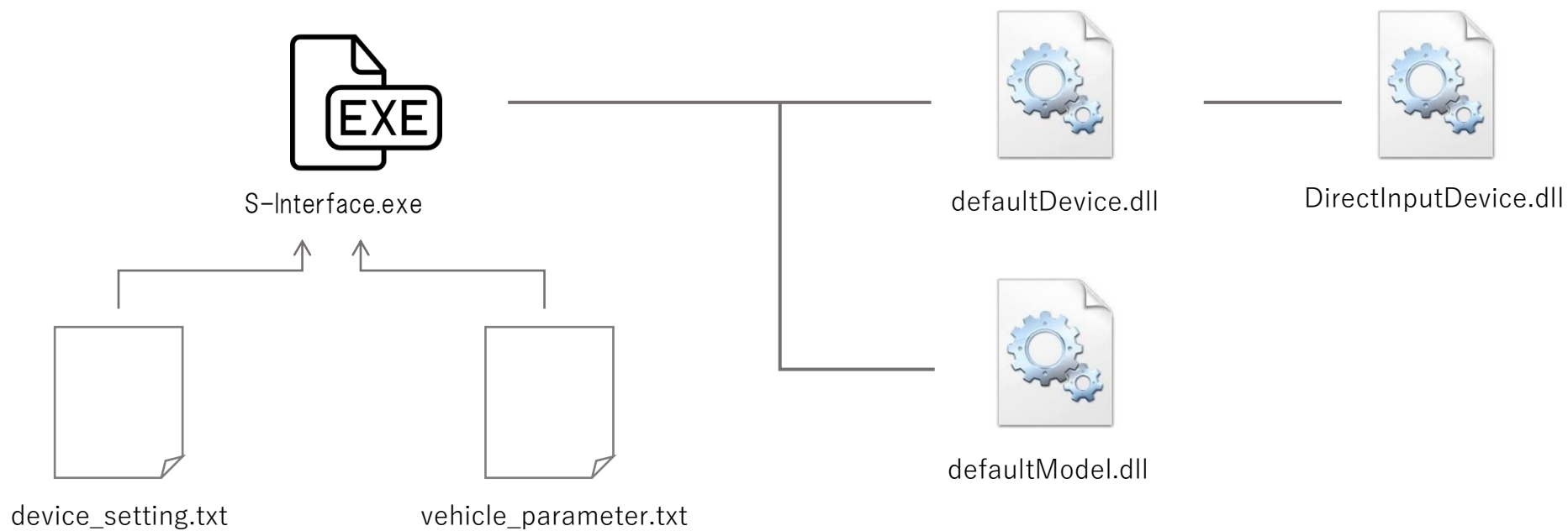
UE4は、初回の実行では時間が掛かることがあり、大きめの数値を設定しています。

また、Simulinkを使うときなどは、Simulinkの立ち上がりに時間が掛かることがあるので、適宜調整します。

4. S-Interface

S-Interfaceのプログラムとして、通常のゲームハンドル対応のS-Interface.exeと、Simulink対応のS-Interface-Simulink.exeの2種類があります。

通常のゲームハンドル対応のS-Interface.exeは、sbin/S-Interfaceにある device_setting.txt と vehicle_parameter.txt の2つの設定ファイルを読み込みます。また、defaultDevice.dll と defaultModel.dll の2つのDLLを使用します。Forcefeedbackデバイスを使う場合には、さらに、DirectInputDevice.dll を使用します。



```
device_setting.txt - メモ帳
ファイル(F) 編集(E) 書式(O) 表示(V) ヘルプ(H)
=====
#
#           Device Setting
#
# Format:
# Input Name, Device Data, Factor, Zero
#
#   Value = Factor * (Device Data - Zero)
#
# Device Data would be ...
#   xPos, yPos, zPos, uPos, vPos, rPos
#
#=====
# Uncomment following line to show data on console
#Show Row Data
#Show State Data
#Show Data

# Use DirectInput Device
Device_DLL, C:\MDS-series\Sirius\sbin\S-Interface\DirectInputDevice.dll
Init Func, initializeDIdevice
Interact Func, getInfoAndSetForce
Term Func, releaseDIdevice
ForceFB Device, no

# Steering in [rad]
Steer,xPos,2.396737e-4,32767

# Accel Pedal
Accel,zPos,-2.396737e-4,32767

# Brake Pedal
Brake,zPos,2.396737e-4,32767

# Headlight ON SW
HeadLightON,2

# Headlight OFF SW
HeadLightOFF,1

# Winker Right SW
WinkerRight,32

# Winker Left SW
WinkerLeft,16

# Move Direction(Fore/Back) Change SW
MoveDirectionChange,4
```

device_setting.txt の内容を左に示します。

#で始まる行はコメント行です。

Show * Dataのコメントを外すと、S-Interfaceのコンソールに、ゲームハンドルから取得した情報が表示されます。
新たに設定が必要な場合は、その表示を見ながら、軸やスイッチの対応などを設定します。

ForceFeedbackデバイスを使用する場合は、ForceFB Device を yes とします。

関数名などは変更する必要はありません。

デバイスによっては、この部分があると、正常に認識できない場合があるので、その場合はコメントにしておきます。

ハンドル、アクセル、ブレーキが、どの軸(xPos, yPos, zPos, uPos, vPos, rPos)であるか、また、他に変換のための係数と、ゼロ点位置を設定します。左の例では、

$$\text{舵角[rad]} = (\text{xPosの値} - 32767) \times 2.396737\text{e-4}$$

$$\text{アクセルストローク量} = (\text{zPosの値} - 32767) \times (-2.396737\text{e-4})$$

$$\text{ブレーキストローク量} = (\text{zPosの値} - 32767) \times (2.396737\text{e-4})$$

ヘッドライト、ウィンカー、前進/後進の切り替えを、ゲームハンドルのどのスイッチで行うかを設定します。

設定する値は、上のShow Row Dataなどを有効にして、コンソールに値を出して、ボタンを押して、スイッチの値がいくつになっているかを見て、決定します。

```

vehicle_parameter.txt - メモ帳
ファイル(F) 編集(E) 書式(O) 表示(V) ヘルプ(H)
#=====
#           Vehicle Parameter Setting
#=====

# Uncomment Following to Output CSV Log File
#Output CSV Log

#
# Parameters
#
AccelCreep in [G], 0.04
DragCoef in [G / (m/s)^2], 0.0002
InverseSteerRatio, 0.08333
YawMomentIntertia [kgm^2], 2742.0
FrontCorneringPower [N/rad], 68909.0
RearCorneringPower [N/rad], 51406.0
Distance CGtoFrontWheel [m], 1.0
Distance CGtoRearWheel [m], 1.67
FrontTread [m], 1.59
RearTread [m], 1.59
Mass [kg], 1800.0
MaxVCreep [m/s], 1.5
RollingResist [G], 0.05
RollGain [rad/G], 0.05
PitchGain [rad/G], 0.01
AccelCutoffFreq [rad/s], 5.0
BrakeCutoffFreq [rad/s], 5.0
RollCutoffFreq [rad/s], 3.0
PitchCutoffFreq [rad/s], 0.5
SusCutoffFreq [rad/s], 8.0
SusDampingCoef [rad/s], 0.6
AccelGain [-], 0.4
BrakeGain [-], 0.7
SteerGain [-], 1.0
MaxReactionForceAngle [rad], 3.141592
LowSpeedReactionForceGain [-], 0.25
LowSpeedLimitOfReactionForce [km/h], 20.0
HighSpeedLimitOfReactionForce [km/h], 60.0

```

vehicle_parameter.txt の内容を左に示します。

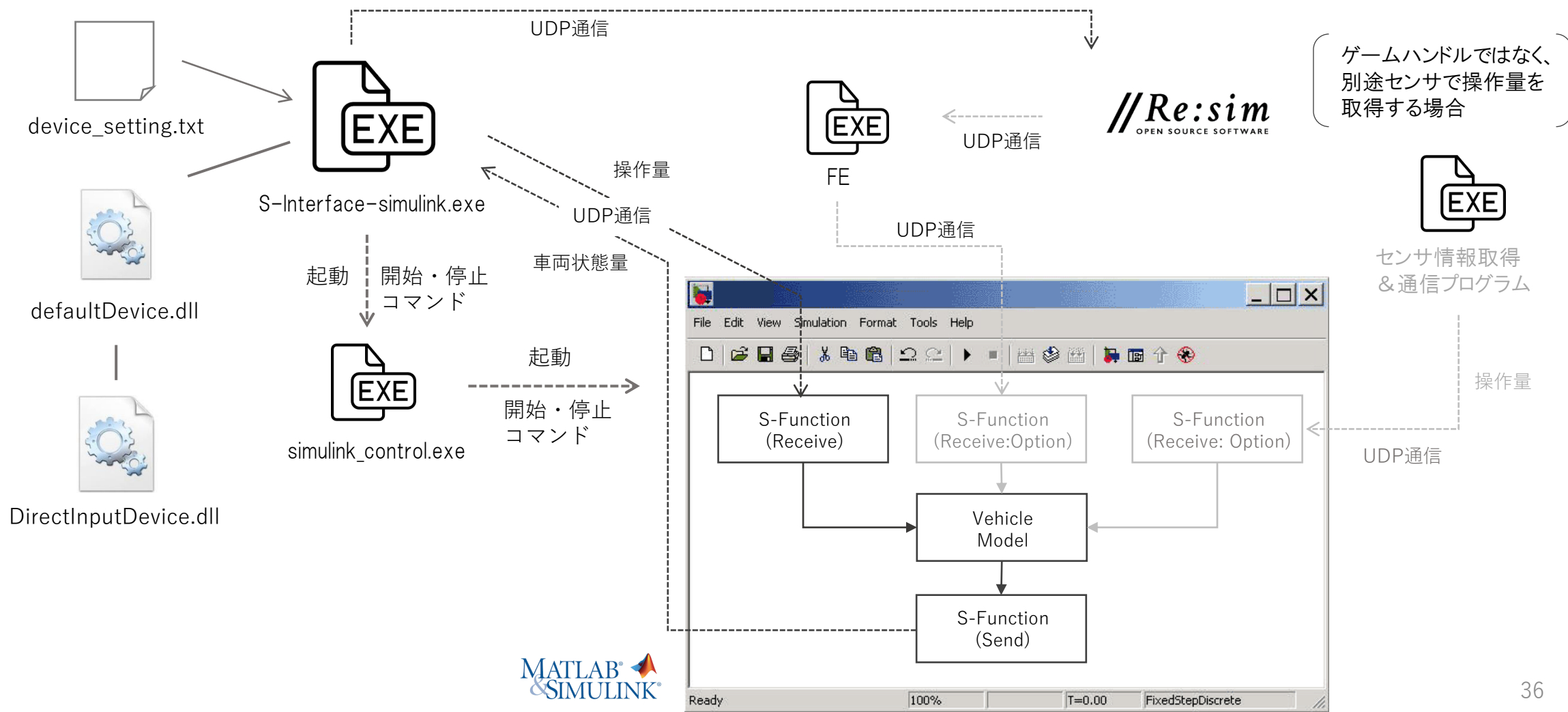
#で始まる行はコメント行です。

タグ名を見れば、大体何のパラメータであるか分かると思うので、詳しくは省略します。

Simulink対応のS-Interface-simulink.exeは、sbin/S-Interfaceにある device_setting.txt を読み込みます。また、ゲームハンドルを扱うために defaultDevice.dll を使用します。Forcefeedbackデバイスを使う場合には、さらに、DirectInputDevice.dll を使用します。

車両モデル部分はSimulink側に実装し、計算結果をS-Interface-simulinkへ送信するようにします。

Simulinkで他車両の情報を必要とする場合には、FEを作り、Re:simから情報を中継して、Simulinkへ送る仕組みとします。



5. Re:simとの連携

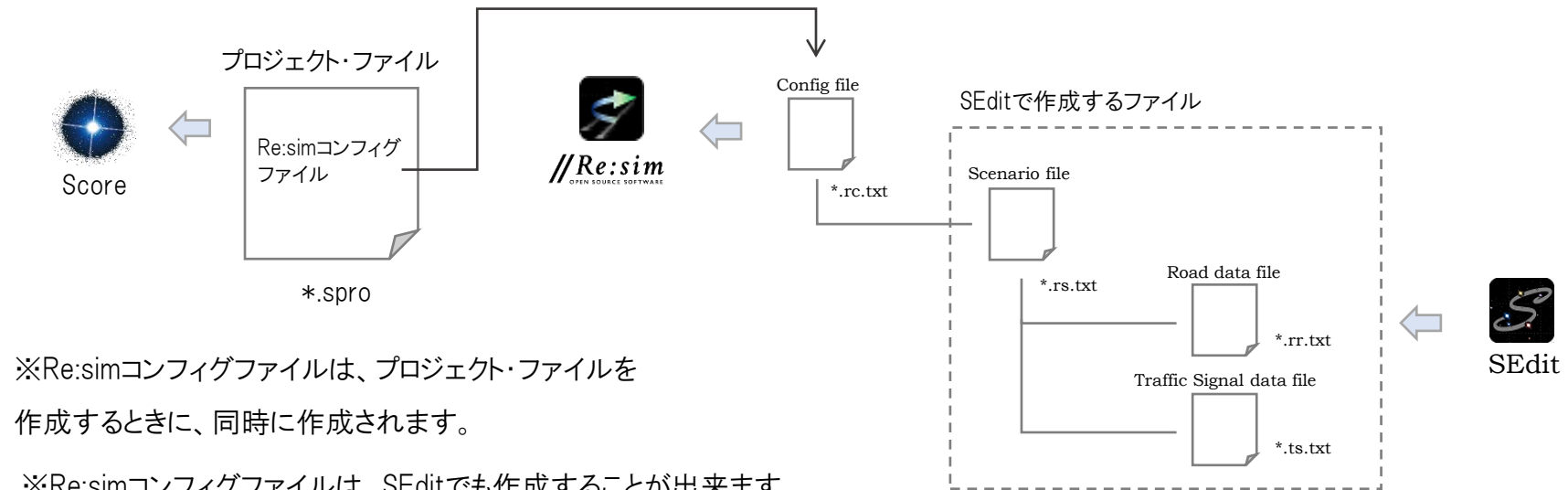
交通流シミュレーションのための道路データやOD交通量設定、シナリオ作成はSEditにより行います。

その詳細については、SEdit-Manual-Part1-Basic_Operation.pdf と SEdit-Manual-Part2-Scenario_Data_Setting.pdf を参照してください。

SEditで作成したRe:sim用のシミュレーションデータをSiriusで使用するための作業は、プロジェクトを作成するだけです。

プロジェクトの作成方法については、「3.4 プロジェクトの編集」を参照してください。

ここでは、ファイル依存関係を示しておきます。

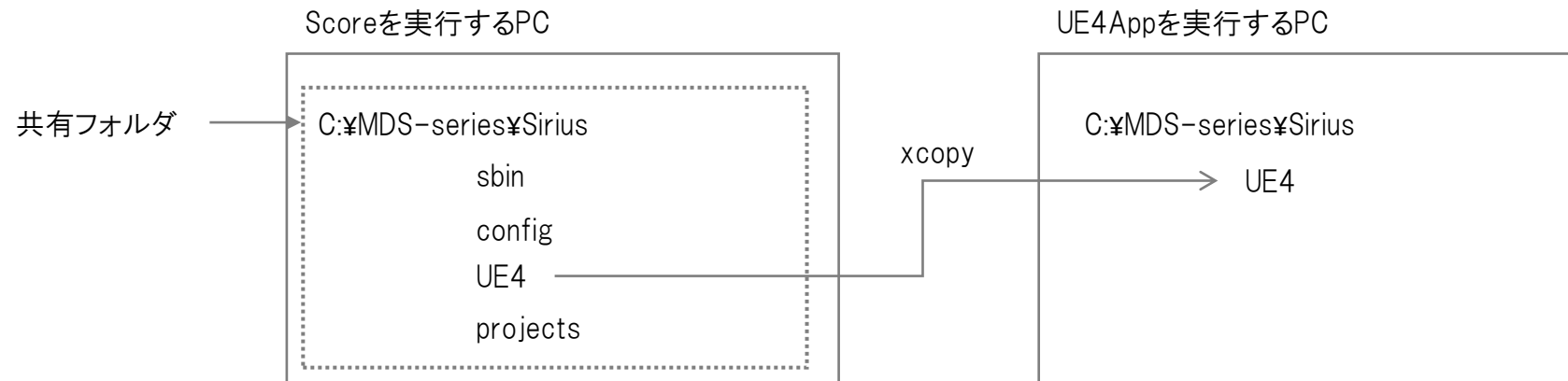


※Re:simコンフィグファイルは、プロジェクト・ファイルを作成するときに、同時に作成されます。

※Re:simコンフィグファイルは、SEditでも作成することができます
(Re:sim単独でシミュレーションを行うこともあるため)

6. UE4との連携

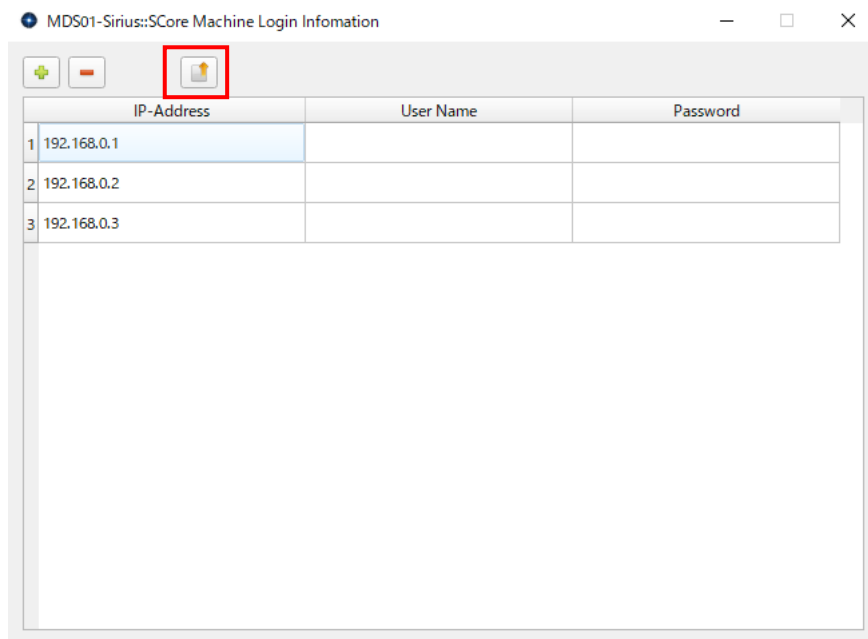
UE4のプログラム・データは、サイズが非常に大きくなるので、毎回実行時にネットワーク・ドライブから持ってくるのは非効率的になります。そこで、UE4のプログラム・データだけは、UE4Appを実行するPCのローカルフォルダにコピーしておきます。



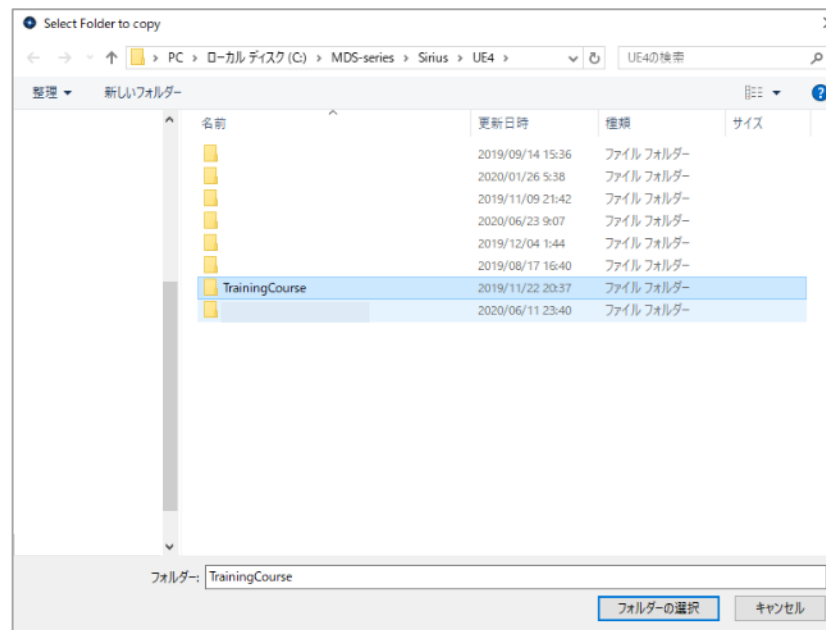
そのために、ログイン情報編集ダイアログにある配信ボタンを使用します。

配信ボタンを押すと、UE4フォルダ内のどのフォルダをコピーするか選択するダイアログが表示され、次に、どのPCにコピーするかを選択し、PMTtoolsのPsExecを使って、リモートPC側でxcopyコマンドを実行して、フォルダをコピーします。

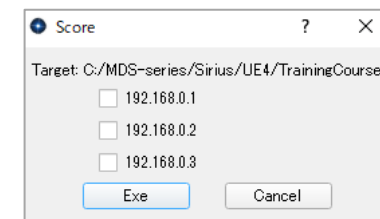
①配信ボタンを押す



②コピーするフォルダの選択



③コピー先のPCを選択

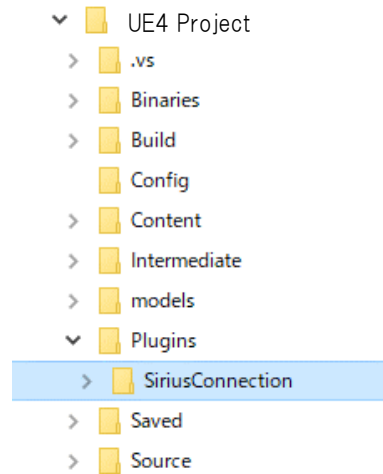


※Sirius/UE4もしくは、それより下にあるフォルダ
以外を選択すると、次のステップに進みません。

UE4との連携は、Siriusプラグインを介して行います。

SiriusConectionを、UE4のプロジェクトのPluginsフォルダにコピーします。

これによって、UE4のブループリント(BP)にて、Sirius関連のブロックを使用できるようになります。

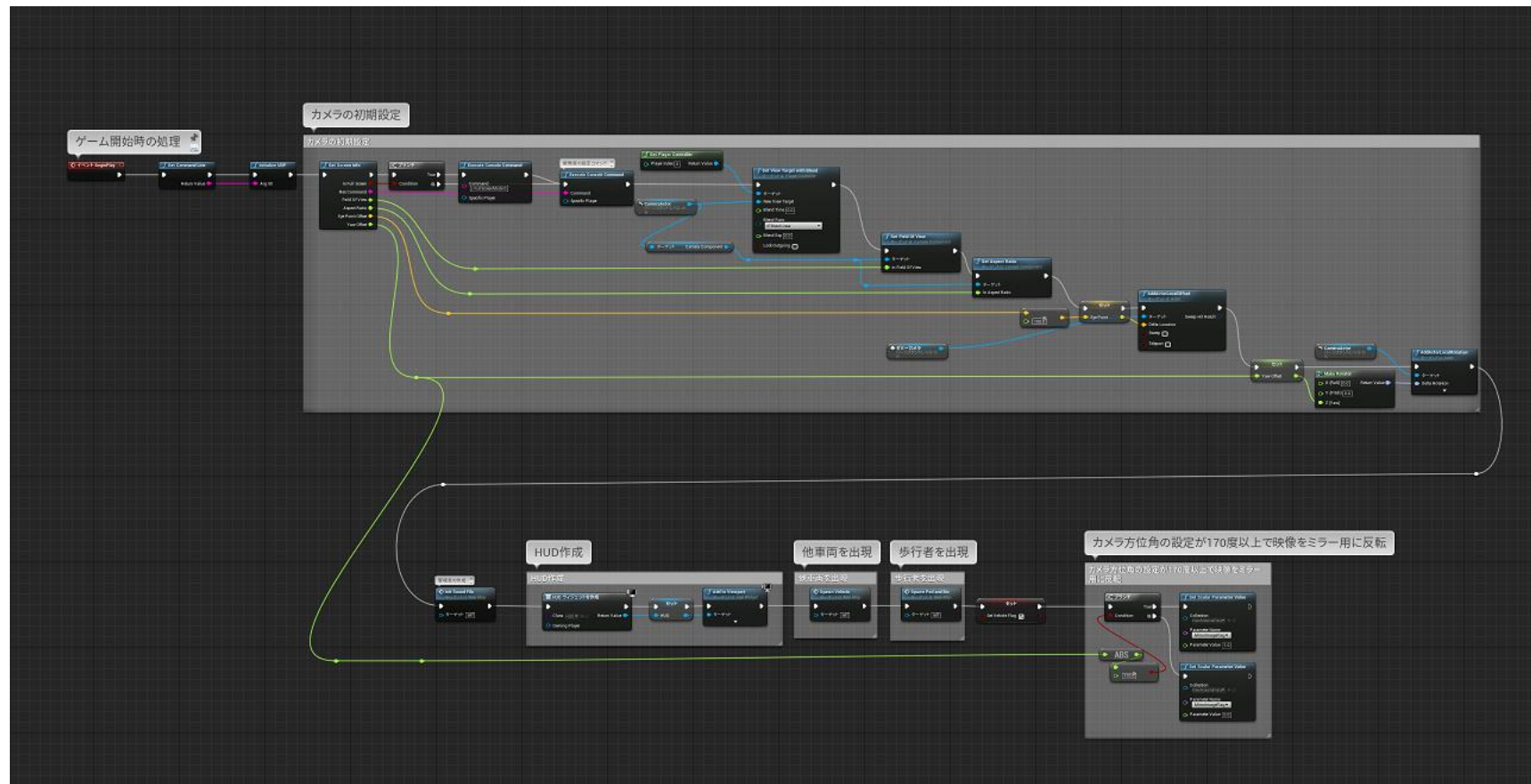


UE4プロジェクトを1から作るのは結構大変なので、既存のプロジェクトのコピーを作り、それを編集するか、コンテンツをマイグレーションし、BPもコピー & ペーストして移植するなどするのが簡単です。

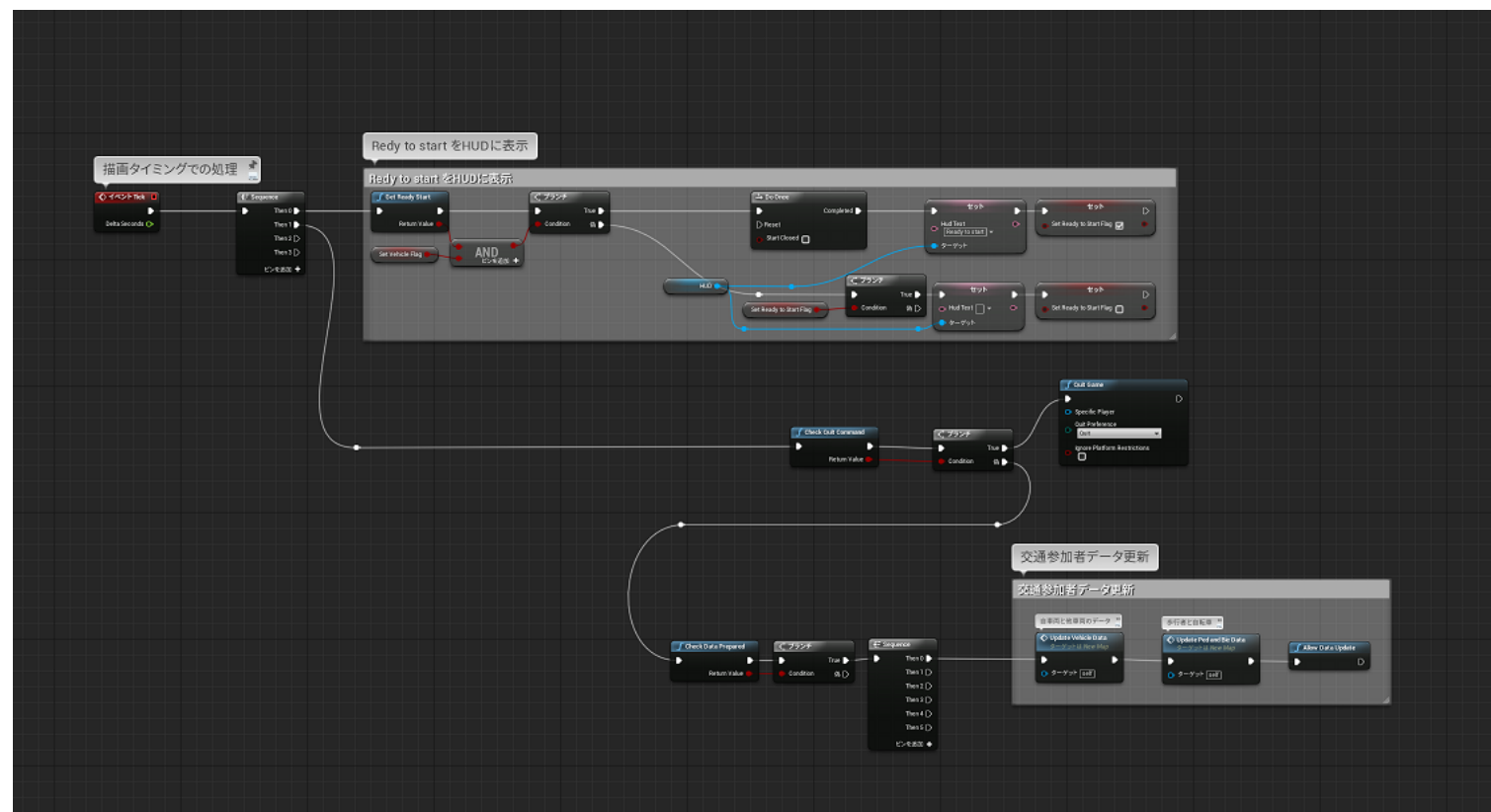
以下では、BPを掲載していますが、紙面の大きさ上、すべてを掲載できないので、実際についてはUE4プロジェクトを参照してください。

必ず設定しなければならない初期化に関するBPは以下ようになります。

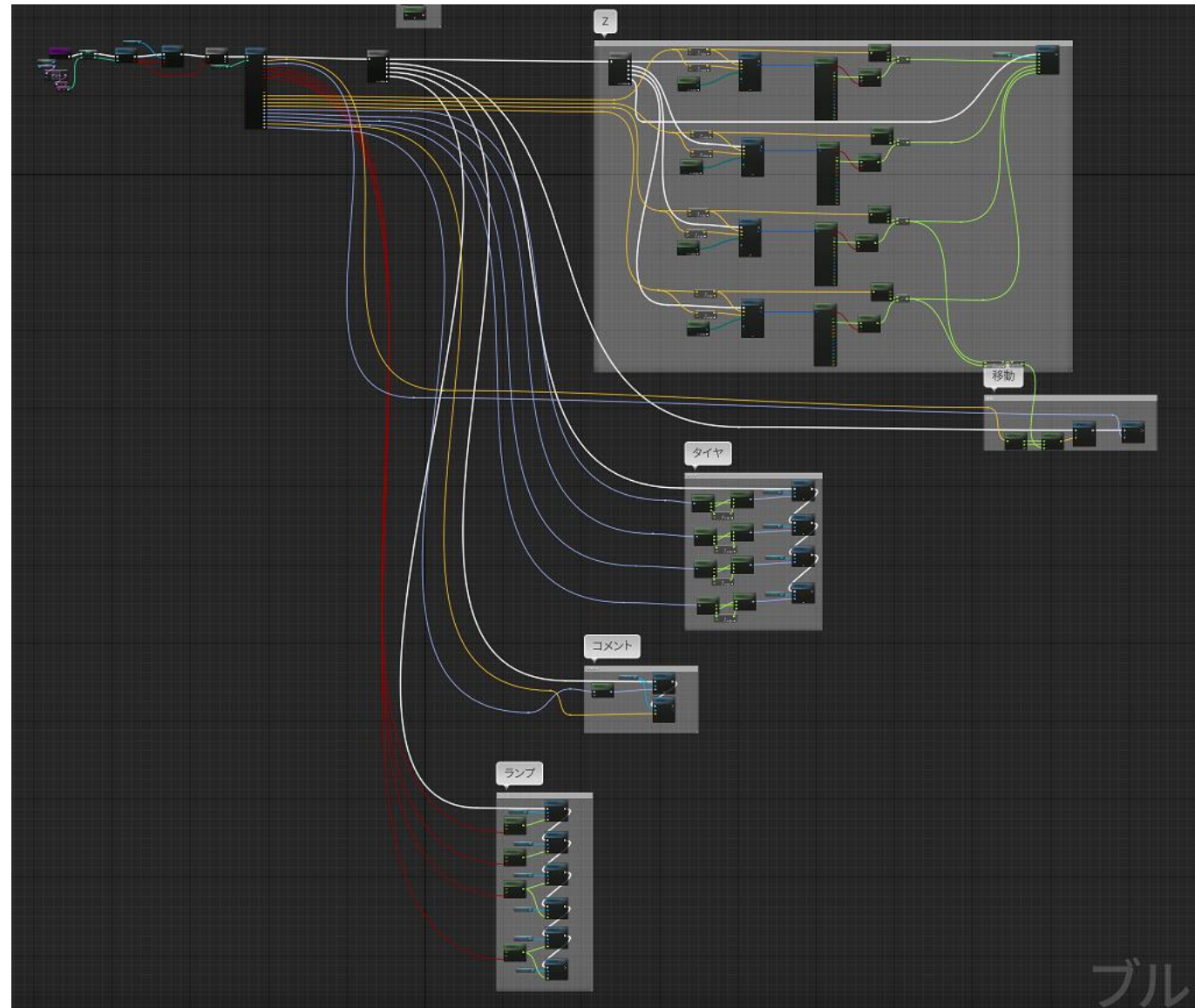
レベルBPのbeginPlayイベントに実装します。



S-Interface, Re:sim, FEなどによって、アクターを動かしたりするBPは、以下のように、レベルBPのtickイベントとして実装します。



車両1台につき、以下のようなBPを組むことで、車両位置の移動や高さ調整、タイヤの回転、ランプ類の点灯などを制御します。

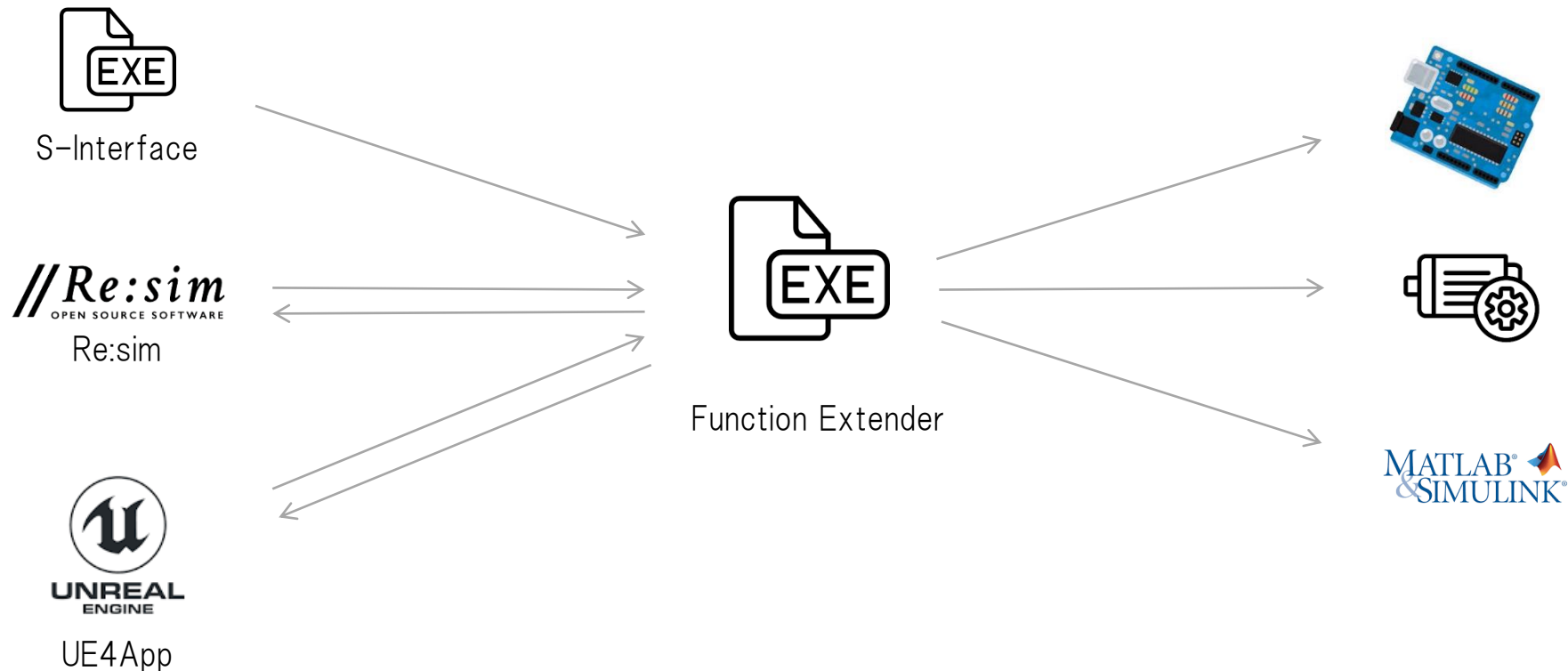


7. Function Extender

Function Extenderは、シミュレーションを機能拡張するためのプログラムです。

S-InterfaceやRe:simからデータを受け取って、何らかの処理を行って、Re:simやUE4App、または外部接続機器へ信号を送るなどの使い方をします。

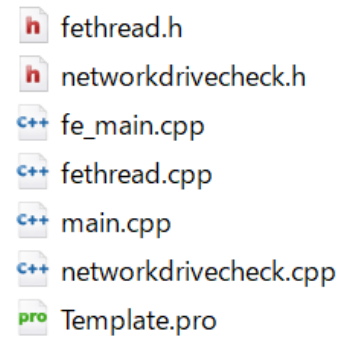
したがって、プログラムの的には、「データの受信—処理—データの送信」を繰り返し行うシンプルなものです。



Function Extenderは、コンソール・プログラムで、テンプレートとなるソースコードを提供しています。

開発環境は、Qt5 です。(UDP通信さえできれば、言語は何でも構いません。テンプレートを移植してください。)

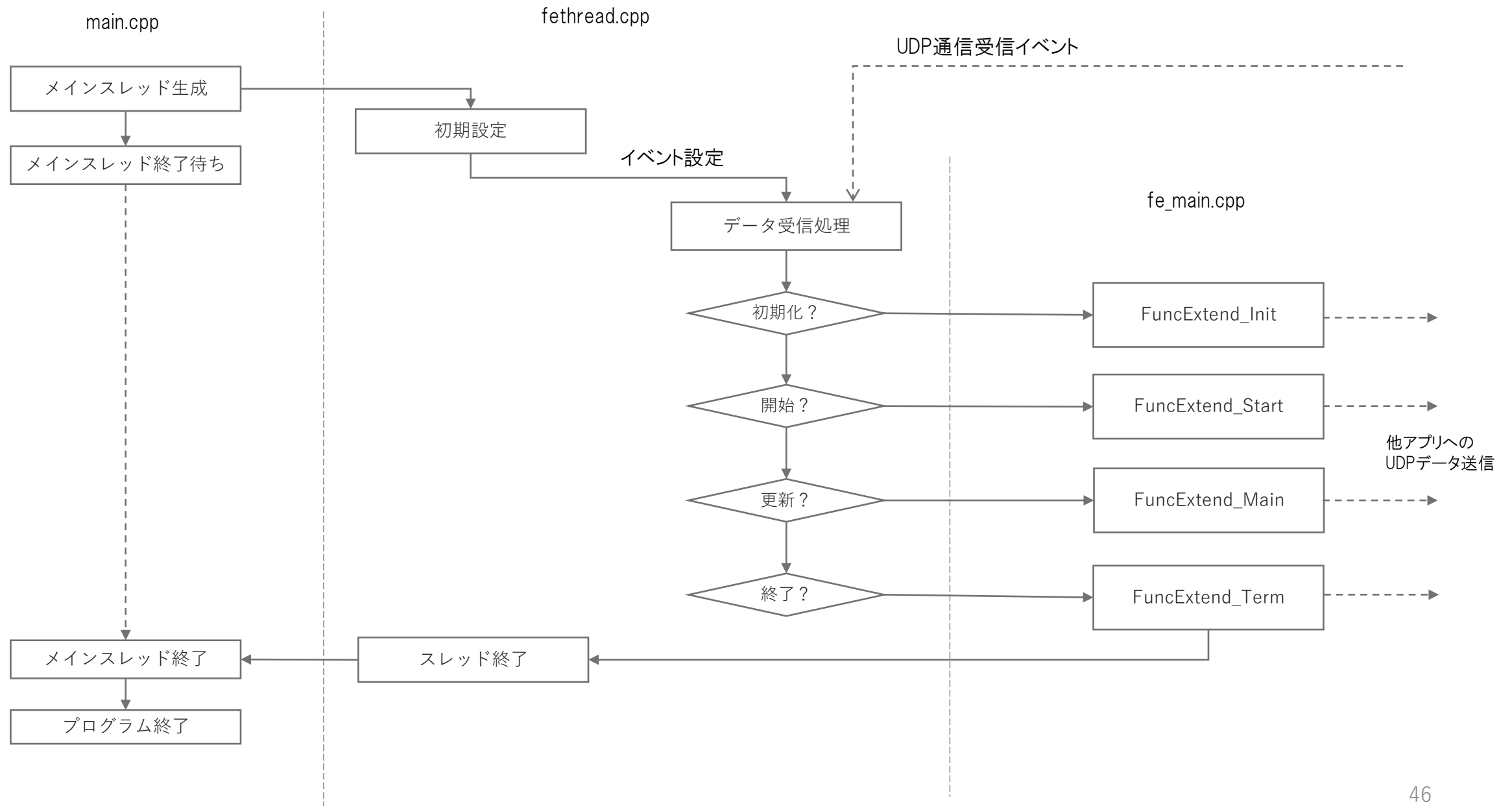
Template



簡単なフローチャートを次ページに示します。

基本的には、FuncExtend_Init(), FuncExtend_Start(), FuncExtend_Main(), FuncExtend_Term()の4つの関数を、実現したいことに合わせてコーディングすることになります。

それ以外のデータ取得部分などは、テンプレートのコードのままで、通常は問題ありません。

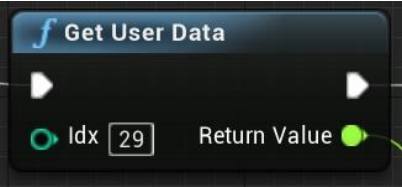


Function Extenderから、Re:sim、UE4などへデータを送る場合は、以下のデータフォーマットで送信します。

なお、送信データ・フォーマットについては、随時見直しを行っており、変更になる場合があります。

宛先	コマンド概要	備考	送信データ バイト数	送信データ																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																											
				0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																
S-Interface	S-Interfaceオブジェクト(DS車両)のワープ	CZ: Z座標の値を変更するかどうか 'y'のときに送信データのZを使う 座標はUE4の座標系で指定し、ヨー角は[deg]	19	'F'	'W'	float X				float Y				float Yaw				CZ	float Z																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																												
	S-Interfaceオブジェクト(DS車両)のヘッドライト状態変更	HL = 0(OFF), 1(Low), 2(High) この値はS-InterfaceからUE4Appに送信される	7	'F'	'H'	'L'	int HL																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																								
	S-Interfaceオブジェクト(DS車両)の自動運転制御コマンド	controlMode=0, 1, 2, 3 0: 通常の手動運転 1: アクセル、ブレーキ、ハンドル舵角のコマンドを使用、トルクは発生しない 2: アクセル、ブレーキコマンドを使用、トルクでハンドル動かし、その値が車両モデルに入る 3: アクセル、ブレーキ、ハンドル舵角コマンドを使用、トルクでハンドル動くが、舵角は制御には使用しない	23	'F'	'A'	'C'	int controlMode				float accelCom				float brakeCom				float steerCom				float trqCom																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																								</

* UE4Appへのユーザーデータは、Siriusプラグインの
Get User Dataブロックで、BPで参照することができます。
(index番号を指定します)



宛先	コマンド概要	備考	送信データ バイト数	送信データ																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																											
				0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																
Re:sim	信号現示の変更	targetTSID : 対象の信号機のID targetColor: 0 : 次の信号現示に遷移 その他は以下の通り TRAFFICSIGNAL_BLUE 1 TRAFFICSIGNAL_YELLOW 2 TRAFFICSIGNAL_RED 4 TRAFFICSIGNAL_RIGHT_RED 12 TRAFFICSIGNAL_STRAIGHT_RED 20 TRAFFICSIGNAL_STRAIGHT_RIGHT_RED 28 TRAFFICSIGNAL_LEFT_RED 36 TRAFFICSIGNAL_STRAIGHT_LEFT_RED 52 TRAFFICSIGNAL_YELLOWFLUSH 64 TRAFFICSIGNAL_REDFLUSH 128 duration : 信号現示の表示時間[s]	15	'F'	'T'	'c'	int targetTSID				int targetColor				float duration																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																