



SCENARGIE®

Scenargie®2.1 インストレーションガイド

Space-Time Engineering, LLC

2016 年 9 月

目次

はじめに.....	1
1. 準備.....	2
1.1. 用語.....	2
1.2. 動作環境.....	2
1.3. ユーザ登録.....	4
1.4. プロダクトの入手.....	5
2. LS の設定.....	6
2.1. Windows での LS の設定.....	6
2.1.1. ライセンス管理プログラムの入手.....	6
2.1.2. ライセンス管理プログラムのインストールと起動.....	7
2.1.3. LS のアップデート.....	8
2.2. Linux での LS の設定.....	9
2.2.1. ライセンス管理プログラムの入手.....	9
2.2.2. ライセンス管理プログラムのインストールと起動.....	9
2.2.3. LS のアップデート.....	11
2.3. Mac OS X での LS の設定.....	12
2.3.1. ライセンス管理プログラムの入手.....	12
2.3.2. ライセンス管理プログラムのインストールと起動.....	12
2.3.3. LS のアップデート.....	14
2.4. ライセンスの設定方法.....	15
2.4.1. LS の登録.....	15
2.4.2. LS へのライセンストークンの移動.....	15
2.4.3. LS から SS へのライセンスの返却.....	16
2.5. LS の変更.....	17
2.6. 新規ライセンスの追加手順.....	17
3. Scenargie のインストール.....	18
3.1. Windows 環境へのインストール.....	18
3.1.1. Scenargie Visual Lab のインストール.....	18
3.1.2. Scenargie Base Simulator のインストール.....	20
3.1.3. Visual C++ のインストール.....	21
3.1.4. Scenargie Base Simulator のビルド.....	22
3.1.5. Scenargie 稼働コンピュータでのライセンスの有効化.....	24
3.2. Linux 環境へのインストール.....	25
3.2.1. Scenargie Visual Lab のインストール.....	25
3.2.2. Scenargie Base Simulator のインストール.....	27

3.2.3.	Scenargie 稼働コンピュータでのライセンスの有効化	30
3.3.	Mac OS X 環境へのインストール	31
3.3.1.	Scenargie Visual Lab のインストール	31
3.3.2.	Scenargie Base Simulator のインストール	33
3.3.3.	Scenargie 稼働コンピュータでのライセンスの有効化	36
3.4.	ビルドオプション一覧	37
3.4.1.	全モジュール共通のビルドオプション	37
3.4.2.	Multisystems で使用可能なビルドオプション	37
4.	オンラインによるライセンス認証	38
4.1.	オンラインチェックアウトの手順	38
4.1.1.	環境変数の設定	38
4.1.2.	ユーザ認証	40
5.	オフラインによるライセンス認証	42
5.1.	オフラインチェックアウトの操作手順(Scenargie 稼働コンピュータでの操作その 1)	42
5.2.	オフラインチェックアウトの操作手順(LS または SS での操作)	43
5.2.1.	LS からオフラインチェックアウトする場合	43
5.2.2.	SS からオフラインチェックアウトする場合	43
5.3.	オフラインチェックアウトの操作手順(Scenargie 稼働コンピュータでの操作その 2)	43
5.3.1.	環境変数の設定	44

はじめに

本書は、離散事象シミュレータ Scenargie2.1 のインストール手順を示すものです。

関連ドキュメント一覧

インストレーションガイド
プログラマーズガイド
Visual Lab ユーザガイド
Base Simulator ユーザガイド
Base Simulator モデルリファレンス
Dot Eleven Module ユーザガイド
Emulation Module ユーザガイド
LTE Module ユーザガイド
ITS Extension Module ユーザガイド
Multi-Agent Extension Module ユーザガイド
Multi-Agent Extension Module モデルリファレンス
Fast Urban Propagation Module ユーザガイド
High Fidelity Propagation Module ユーザガイド
Trace Analyzer ユーザガイド

1. 準備

1.1. 用語

本ドキュメントでは以後、以下の用語を使用します。

- SS (STE hosted license server)
Space-Time Engineering, LLC (以降、STE と呼ぶ) がホストし、インターネットを経由して Scenargie のライセンス認証を行うサーバです。
- LS (Local license server)
ユーザがホストし、LAN を経由して Scenargie のライセンス認証を行うサーバです。
LS の設定変更は、LS 用ウェブ UI から行います。
- ライセンス管理プログラム(Scenargie License Manager)
LS、インストールプログラムを含むライセンスプログラム一式です。
- アカウント
ライセンスのご購入者(組織)に対して発行され、購入したライセンスを保有します。各ライセンスの利用者(ユーザ)は、アカウントに属することで、そのアカウントが保有するライセンスを使用することが可能です。
- ユーザ権限
アカウントに属するユーザは、Owner、Admin、Member のいずれかのユーザ権限を持ちます(詳細は、STE ウェブサイト内 My accounts ページをご参照ください)。尚、アカウント作成時は、Owner 権限を持ったユーザ 1 名のみがアカウントに属します。

1.2. 動作環境

1) ソフトウェア条件

Windows

Windows 7 32bit	Visual Studio Express 2013 for Windows Desktop Update 5
Windows 7 64bit	Visual Studio Express 2013 for Windows Desktop Update 5
Windows 10 64bit	Visual Studio Express 2015 for Desktop Update 1

Linux

Red Hat Enterprise Linux 6 64bit	gcc 4.8.2、glibc 2.12、GNOME
Red Hat Enterprise Linux 7 64bit	gcc4.8.5、glibc 2.17、GNOME
CentOS 6 64bit	gcc 4.8.2、glibc 2.12、GNOME
CentOS 7 64bit	gcc 4.8.5、glibc 2.17、GNOME

Mac OS X

Yosemite 10.10.5 64bit	Xcode 7.2.1、Apple LLVM version 7.0.2 (clang-700.1.81)
El Captitan 10.11.3 64bit	Xcode 7.2.1、Apple LLVM version 7.0.2 (clang-700.1.81)

注意) Emulation Module は Linux のみのサポートとなります。

注意) Fast Urban Propagation Module 及び High Fidelity Propagation Module は Linux 64bit と Windows 64bit のみのサポートとなります。

注意) Red Hat Enterprise Linux 6 及び CentOS 6 で Scenargie を使用する場合、gcc 4.8 を含む Developer Toolset が必要です。

以下のコマンドにより gcc 4.8 の Developer Toolset をインストールしてください。

```
# wget http://people.centos.org/tru/devtools-2/devtools-2.repo -O
/etc/yum.repos.d/devtools-2.repo
# yum install devtoolset-2-gcc-c++ devtoolset-2-binutils
```

上記コマンドにより Developer Toolset をインストールした後、以下のコマンドで Developer Toolset を有効にすることで、gcc 4.8 によるビルドを行えるようになります。

```
$ scl enable devtoolset-2 bash
```

元の環境に戻したい場合は exit により Developer Toolset 環境から離脱します。

```
$ exit
```

また、以下のようにビルド時のみ Developer Toolset を有効にすることも可能です。

```
$ scl enable devtoolset-2 "make -f makefile.linux"
```

注意) Mac OS X 環境で Scenargie を使用する場合、Command Line Tools が必要です。

以下のコマンドにより Command Line Tools をインストールしてください。

```
$ xcode-select --install
```

2) ハードウェア推奨条件

Linux、Windows、Mac OS X

CPU: 1.8GHz 以上

Memory: 1GB 以上

HDD: 50GB 以上

Display: 解像度 1024 × 768 以上

3) SS および LS のサポートブラウザ

Windows7、Windows10

Internet Explorer 10 以降

Google Chrome version39 以降

Mozilla Firefox version31 以降

Linux(CentOS6、CentOS7)

Google Chrome version39 以降

Mozilla Firefox version31 以降

Mac OS X

Yosemite Safari 8

El Capitan Safari 9

注意) Red Hat Enterprise Linux 6 及び CentOS 6 でライセンスサーバを稼働する場合、Ishal コマンドが必要になります。

Ishal コマンドが入っていない場合は、以下のコマンドによりインストールしてください。

```
# yum install hal
# /etc/init.d/haldaemon start
```

注意) Red Hat Enterprise Linux 7 及び CentOS 7 でライセンスサーバを稼働する場合、dmidecode コマンドが必要になります。

dmidecode コマンドが入っていない場合は、以下のコマンドによりインストールしてください。

```
# yum install dmidecode
```

1.3. ユーザ登録

プロダクトの入手及びライセンスを使用するには、STE のウェブサイト (<https://www.spacetime-eng.com/>) よりユーザ登録を行い、アカウントに属している必要があります。ユーザ登録及びアカウントへの所属が行われていない場合は、以下の手順でユーザ登録とアカウントへの所属を行ってください。

尚、Owner ユーザの場合はライセンス購入時にユーザ登録及びアカウントへの所属が行われておりますので、以下の手順を行う必要はありません。

- 1) STE のウェブサイト (<https://www.spacetime-eng.com/>) にアクセスします。

- 2) Members タブの Member register をクリックし、必要な情報を入力して Register をクリックします。
- 3) 入力したメールアドレスにメールが届くので、Activation Link をクリックして Activate します。
- 4) Members タブに移動し、メールアドレスとパスワードを入力してログインします。
- 5) My accounts タブに移動します。
- 6) Join Account をクリックし、所属するアカウントの Account ID と Owner ユーザのメールアドレスを入力して Send request をクリックします。Account ID および Owner ユーザのメールアドレスは、アカウントの管理者にお問い合わせください。
- 7) Owner ユーザ宛に Activate 用のメールが届きます。Owner ユーザが Activate すると、Activate が完了した旨のメールが届きアカウントへの所属が完了します。

1.4. プロダクトの入手

Scenargie 製品の入手は下記のように行います。

- STE のウェブサイト(<https://www.spacetime-eng.com/>)にアクセスし、ご購入時の登録情報にもとづいて sign in してください。
- 製品ダウンロードページにてパッケージをダウンロードしてください。
- 製品ダウンロードページにてライセンス管理プログラムをダウンロードしてください。

注意) Scenargie を使用するには、STE ウェブサイトでのユーザ登録および所属組織が保有するアカウントへの登録が必要になります。

注意) Mac OS X の Safari 等ウェブブラウザによってはダウンロードファイルを自動的に展開する機能が有効になっている場合があります。

Scenargie では、Visual Lab を除く全てのパッケージで scenargie_simulator というフォルダに展開することを想定していますが、別名のフォルダが自動作成される場合はコマンドラインで各パッケージを展開してください。

なお、以下の操作により自動的に展開する機能を無効にすることができます。

- 「環境設定 (Preference)」→「一般」タブ を開く。
- 「ダウンロード後、“安全な”ファイルを開く」のチェックをはずす。

2. LS の設定

本章は管理者(Owner ユーザおよび Admin ユーザ)向けの内容になっております。

ライセンスのオンラインチェックアウトには、インターネットを経由して SS からライセンスをチェックアウトする方法と、ライセンスを SS から LS に移動後 LAN を経由して LS からライセンスをチェックアウトする方法があります。LS からライセンスをチェックアウトする場合は、事前にライセンスを管理する PC 上で LS の設定が必要になります。SS からライセンスをチェックアウトする場合や、既に LS の設定が行われている場合は、「3. Scenargie のインストール」以降に進んでください。

注意) LS は 7575 番ポートを使用してシミュレータと通信を行っているため、ファイアウォール等をご使用の場合は、外部から 7575 番ポートへアクセスできるように設定を行う必要があります。

注意) LS の時間が実時刻と大きく異なると LS が正常に動作しない場合があります。事前に、ntp 等で正しい時刻を設定してください。

LS は Windows、Linux、Mac OS X の各プラットフォームでの実行に対応しています。

各プラットフォームでの LS 設定方法は、本ドキュメントの以下の節をご参照ください。



- Windows : 2.1.節
- Linux : 2.2.節
- Mac OS X : 2.3.節

2.1. Windows での LS の設定

2.1.1. ライセンス管理プログラムの入手

ライセンス管理プログラムの入手は下記のように行います。

STE のウェブサイト(<https://www.spacetime-eng.com/>)にアクセスし、ご購入時の登録情報にもとづいて sign in してください。

Account Menu 内、LS (Local license server)ページにて、 ボタンを選択します。Scenargie License Manager ウィンドウがポップアップしますので、ダウンロードしたいバージョンの  ボタンから OS を選択してください。OS を選択するとダウンロードが開始されます。

2.1.2. ライセンス管理プログラムのインストールと起動

ここでは、LS 構築方法を説明します。

Windows 版 LS は以下のインストーラを提供しており、インストーラの実行には管理者権限が必要になります。

- 32bit 用 : ScenargieLicenseManager-3-rxxxxx-win32.exe
- 64bit 用 : ScenargieLicenseManager-3-rxxxxx-win64.exe

インストーラを実行して、インストール先を選択し Extract をクリックすることでインストールが行われます。



ここでは LS のインストールを「C:\ScenargieLicenseManager」に行ったものと想定します。

ライセンス管理プログラムの起動には管理者権限が必要になります。ScenargieLS.exe を右クリックし「管理者として実行(A)...」をクリックするか、下記のコマンド例で示すように、管理者権限でコマンドプロンプトを起動し、ScenargieLS.exe の存在するディレクトリに移動して実行してください。

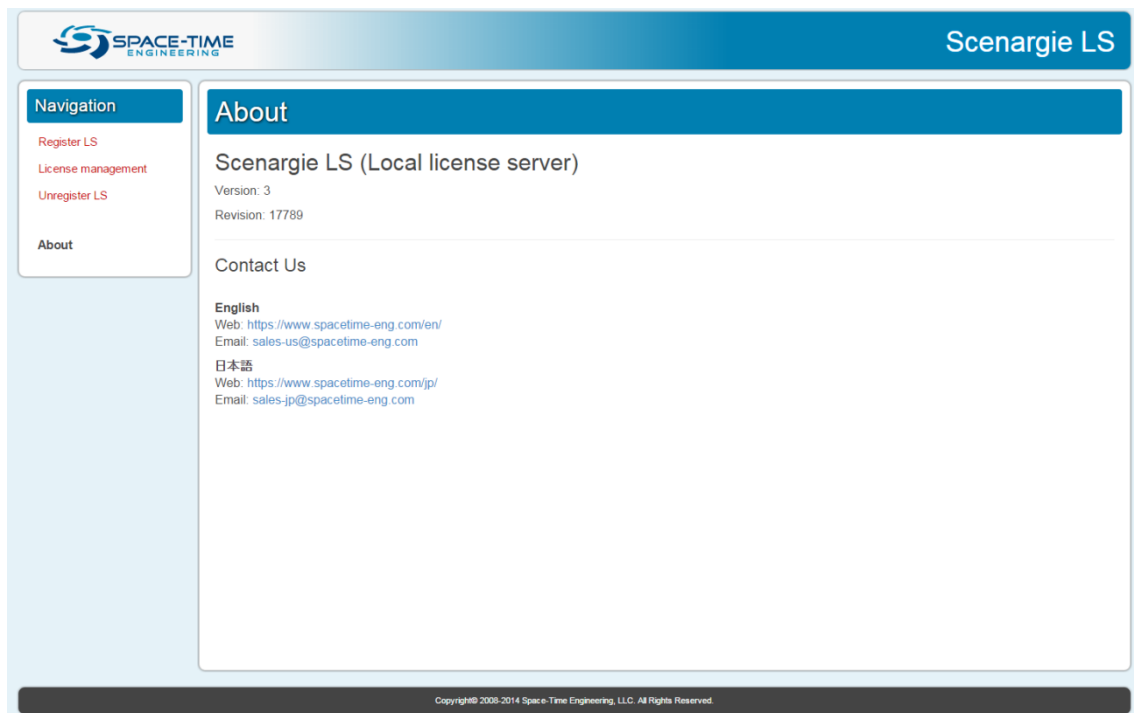
注意) コマンドプロンプトから実行する場合は、必ず ScenargieLS.exe の存在するディレクトリに移動してから実行してください。

<コマンド例>

```
> cd C:\ScenargieLicenseManager
> ScenargieLS.exe
>
```

LS が起動すると、タスクバーに LS が常駐します。LS を終了するには、右クリックメニューより「Exit」を選択してください。尚、コンピュータ起動時に自動的に LS を起動するようにするには、LS 起動後、右クリックメニューより「Enable auto start」を選択してください。

http://localhost:8082 にアクセスし、以下の画面が表示されれば、LS は正常に起動しています。



2.1.3.LS のアップデート



既存の LS をアップデートする場合は、一度 LS を終了し、新しい LS のパッケージを以前の LS と同じ場所にインストールしてください。現在稼働中の LS のバージョンおよびリビジョンは、About 画面の「Version」「Revision」欄に表示されます。

2.2. Linux での LS の設定

2.2.1. ライセンス管理プログラムの入手

ライセンス管理プログラムの入手は下記のように行います。

STE のウェブサイト (<https://www.spacetime-eng.com/>) にアクセスし、ご購入時の登録情報にもとづいて sign in してください。

Account Menu 内、LS (Local license server) ページにて、 ボタンを選択します。Scenargie License Manager ウィンドウがポップアップしますので、ダウンロードしたいバージョンの  ボタンから OS を選択してください。OS を選択するとダウンロードが開始されます。

2.2.2. ライセンス管理プログラムのインストールと起動

Linux 版 LS は以下のインストーラを提供しておりインストーラの実行には管理者権限が必要になります。

- Linux 64bit 用 : ScenargieLicenseManager-3-rxxxxx-linux64.run

下記のコマンド例で示すようにインストーラを実行することにより、LS のインストールが行われます。途中インストール先を入力する箇所がありますので、フルパスで入力してください。

<コマンド例>

```
$ su
# ./ScenargieLicenseManager-3-rxxxxx-linux64.run
Installation path (Full path):
>> /home/user/
# exit
$
```

ここでは LS のインストールをホームディレクトリ直下に行ったものと想定します。

LS は slmd というサービス制御スクリプトで起動や停止を行います。slmd では start、stop、restart、status のコマンドが使用可能です。slmd の実行には管理者権限が必要になります。

<コマンド例>

```
$ sudo service slmd start
```

コンピュータ起動時に LS を自動的に起動する場合は、以下のコマンドで自動起動を有効にします。

<コマンド例>

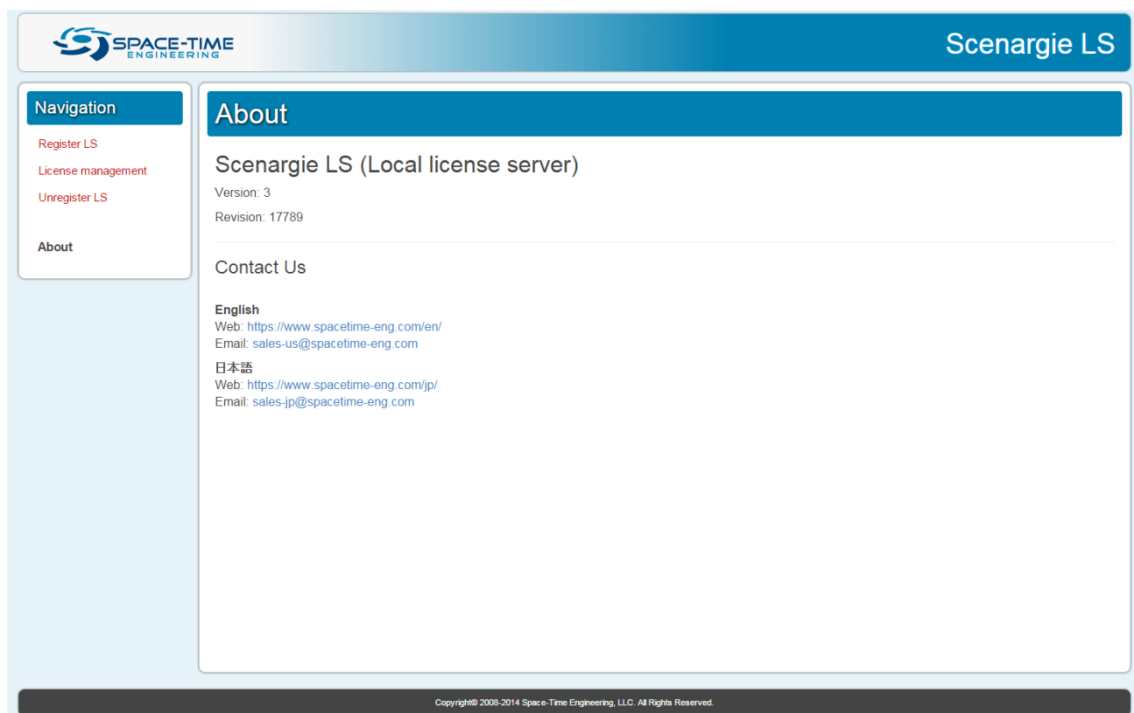
```
$ sudo chkconfig slmd on
```

自動起動を無効にする場合は、以下のコマンドを使用します。

<コマンド例>

```
$ sudo chkconfig slmd off
```

LS 起動後、<http://localhost:8082> にアクセスし、以下の画面が表示されれば、LS は正常に起動しています。



2.2.3.LS のアップデート



既存の LS をアップデートする場合は、一度 LS を終了し、新しい LS のパッケージを以前の LS と同じ場所にインストールしてください。現在稼働中の LS のバージョンおよびリビジョンは、About 画面の「Version」「Revision」欄に表示されます。

2.3. Mac OS X での LS の設定

2.3.1. ライセンス管理プログラムの入手

ライセンス管理プログラムの入手は下記のように行います。

STE のウェブサイト (<https://www.spacetime-eng.com/>) にアクセスし、ご購入時の登録情報にもとづいて sign in してください。

Account Menu 内、LS (Local license server) ページにて、 ボタンを選択します。Scenargie License Manager ウィンドウがポップアップしますので、ダウンロードしたいバージョンの  ボタンから OS を選択してください。OS を選択するとダウンロードが開始されます。

2.3.2. ライセンス管理プログラムのインストールと起動

Mac 版 LS は以下のインストーラを提供しておりインストーラの実行には管理者権限が必要になります。

- ScenargieLicenseManager-3-rxxxxx-mac.run

下記のコマンド例で示すようにインストーラを実行することにより、LS のインストールが行われます。途中インストール先を入力する箇所がありますので、フルパスで入力してください。

<コマンド例>

```
$ su
# chmod +X ./ScenargieLicenseManager-3-rxxxxx-mac.run
# ./ScenargieLicenseManager-3-rxxxxx-mac.run
Installation path (Full path):
>> /home/user/
# exit
```

ここでは LS のインストールをホームディレクトリ直下に行ったものと想定します。

LS は slmd というサービス制御スクリプトで起動や停止を行います。slmd では start、stop、restart、status のコマンドが使用可能です。slmd の実行には管理者権限が必要になります。

<コマンド例>

```
$ sudo service slmd start
```

コンピュータ起動時に LS を自動的に起動する場合は、以下のコマンドで自動起動を有効にします。

<コマンド例>

```
$ sudo cp slmd.plist /System/Library/LaunchDaemons/  
$ sudo launchctl load -w /System/Library/LaunchDaemons/slmd.plist
```

自動起動を無効にする場合は、/System/Library/LaunchDaemons/slmd.plist を開き、以下の行を書き換えます。

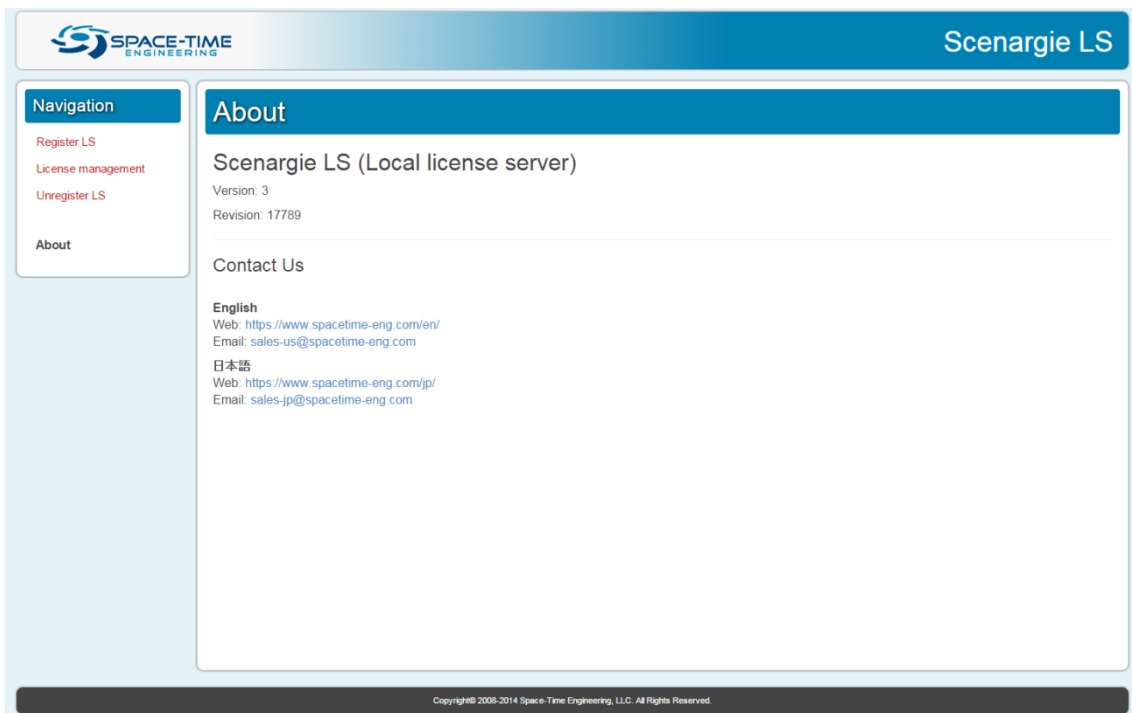
修正前:

```
<key>RunAtLoad</key>  
<true/>
```

修正後:

```
<key>RunAtLoad</key>  
<false/>
```

LS 起動後、<http://localhost:8082> にアクセスし、以下の画面が表示されれば、LS は正常に起動しています。



2.3.3.LS のアップデート

既存の LS をアップデートする場合は、一度 LS を終了し、新しい LS のパッケージを以前の LS と同じ場所にインストールしてください。現在稼働中の LS のバージョンおよびリビジョンは、About 画面の「Version」「Revision」欄に表示されます。

2.4. ライセンスの設定方法



2.4.1. LS の登録

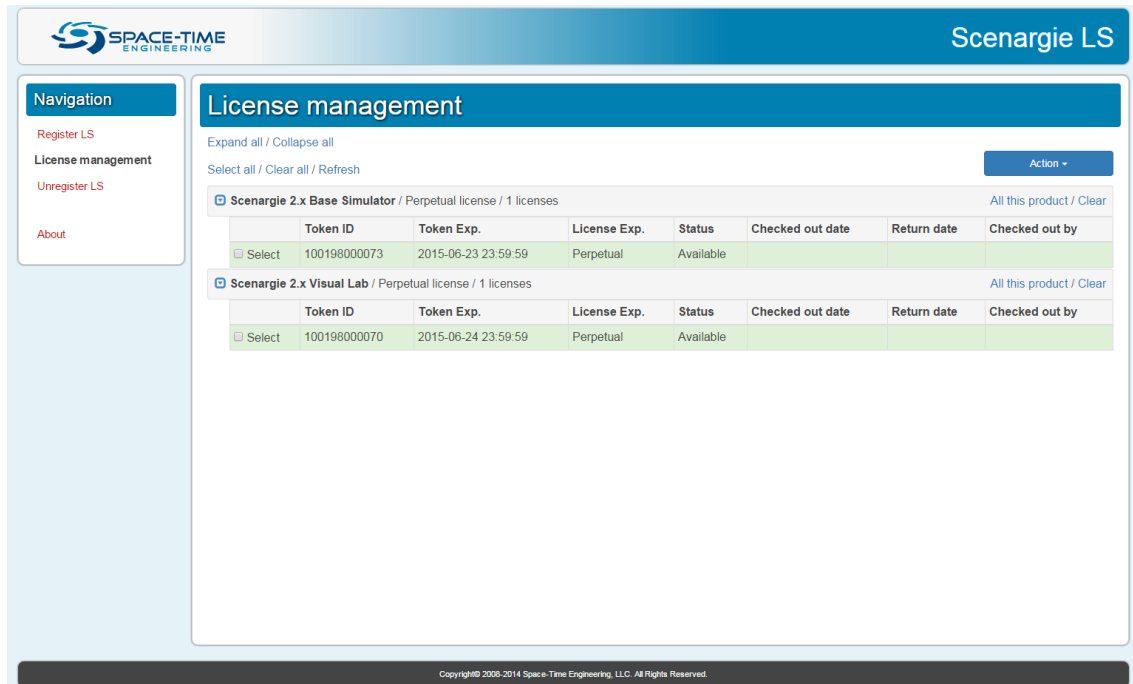
- 1) LS 稼働コンピュータで、ウェブブラウザより <http://localhost:8082> にアクセスします。
- 2) Register LS をクリックします。
- 3)  をクリックし、LS 登録ファイル(ls_registration.dat)を保存します。
- 4) ウェブブラウザより STE のウェブサイト(<https://www.spacetime-eng.com>) にアクセスします。
- 5) 会員ページからログインし、「LS (Local license server)」へ移動します。「5. Give a nickname for your name LS:」の Nickname 欄に LS のコンピュータ名を記載し、3)で作成した LS 登録ファイルをアップロードします。

注意)LS の登録後にハードウェア構成等を変更する場合は、一度 LS の登録を解除し、ハードウェア構成を変更した後に再度 LS を登録する必要があります。

詳細は本ドキュメントの「2.5. LS の変更」をご参照ください。

2.4.2. LS へのライセンストークンの移動

- 1) LS 稼働コンピュータで、ウェブブラウザより STE のウェブサイト (<https://www.spacetime-eng.com>) にアクセスします。
- 2) 会員ページからログインし、「License management」に移動します。
- 3) LS へ移動するライセンスにチェックを入れ、 の「Move selected licenses to LS」を選択し、ライセンスファイル(license_setting.dat)をダウンロードします。
- 4) ウェブブラウザより <http://localhost:8082> にアクセスします。
- 5) License management をクリックします。
- 6)  の「Apply new license」を選択し、ライセンスファイルをアップロードします。




License management に有効なライセンスの一覧が表示されます。

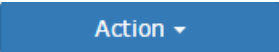
尚、LS へ移動したライセンストークンに対し再度移動の手順を行うことで、移動期間の延長が可能です。

注意) 登録されているアカウントのパスワードを変更した場合、パスワードの変更を反映させるためにライセンスを再度ダウンロードし LS に適用してください。

2.4.3.LS から SS へのライセンスの返却

LS から SS へライセンスを返還する場合は、以下の手順で操作してください。

- 1) LS 稼働コンピュータで、ウェブブラウザより <https://localhost:8082> にアクセスします。
- 2) License management をクリックします。
- 3) SS へ返却するライセンスを選択し、 の「Return selected to SS」を選択し、ライセンス返却ファイル(license_return.dat)をダウンロードします。
- 4) ウェブブラウザより STE のウェブサイト(<https://www.spacetime-eng.com>) にアクセスします。
- 5) 会員ページからログインし、「License management」に移動します。

- 6)  の「Return license to SS」を選択し、ライセンス返却ファイルをアップロードします。

2.5. LS の変更

LS 稼働コンピュータを変更する場合(コンピュータのハードウェア構成を変更する場合も含む)は、一度 LS の登録を解除し、新しいコンピュータ上で再度 LS の登録を行います。

注意) オフラインチェックアウトをしているライセンスが存在する場合は、LS の変更は行えません。

- 1) LS 稼働コンピュータで、ウェブブラウザより <http://localhost:8082> に移動します。
- 2) Unregister LS をクリックします。
- 3)  をクリックし、LS 解除ファイル(ls_unregistration.dat)を保存します。
- 4) ウェブブラウザより STE のウェブサイト(<https://www.spacetime-eng.com>) にアクセスします。
- 5) 会員ページからログインし、「LS (Local license server)」のリンクより LS 解除ファイルをアップロードします。これにより、現在の LS の登録が解除されます。
- 6) 再度、新しいコンピュータで、LS の起動、登録、ライセンスの登録を行います。

2.6. 新規ライセンスの追加手順

モジュールの追加などでライセンスを追加する場合は、以下の手順に従ってください。

- 1) 「2.4.2. LS へのライセンストークンの移動」の手順に従い、新たに追加されたライセンスを LS に移動してください。
- 2) ライセンスファイルを適用します。

3. Scenargie のインストール

本章では、各プラットフォームでの Scenargie のインストールおよびビルドの方法を説明します。
各プラットフォームでのインストール方法は以下の節をご参照ください。

- Windows : 3.1.節
- Linux : 3.2.節
- Mac OS X : 3.3.節

3.1. Windows 環境へのインストール

3.1.1.Scenargie Visual Lab のインストール

パッケージを、任意のディレクトリ以下に展開します。

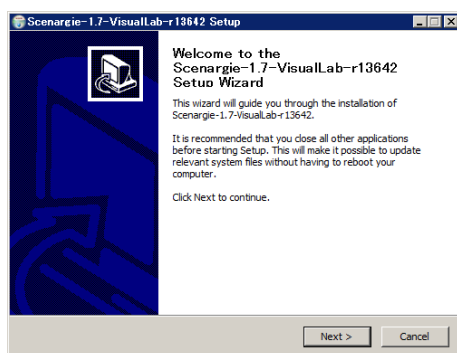
Windows 版のパッケージには下記が含まれます。

- インストーラ(32bit 用) (Scenargie-2.1-VisualLab-rxxxx-32bit-Setup.exe)
- インストーラ(64bit 用) (Scenargie-2.1-VisualLab-rxxxx-64bit-Setup.exe)
- ドキュメント

* 以前のバージョンの Scenargie Visual Lab をアンインストールしてからインストールすることを推奨します。

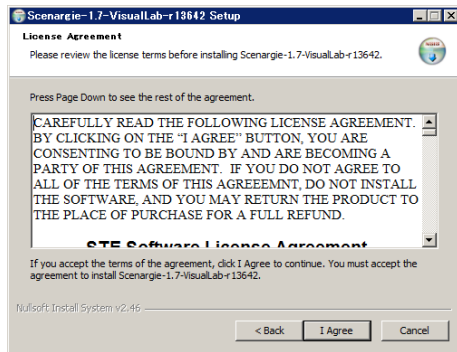
注意) 管理者権限を持たないアカウントでインストールを行う場合、管理者アカウントのパスワード入力を要求されることがあります。

1) インストーラの実行



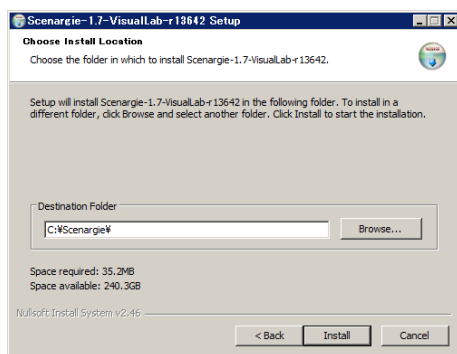
インストーラを起動すると、左の様な画面が表示され、セッアップウィザードが開始されます。

「Next」ボタンをクリックし、先へ進みます。



Scenargie の License Agreement が表示されます。

同意できれば、「I Agree」ボタンをクリックして、次に進みます。

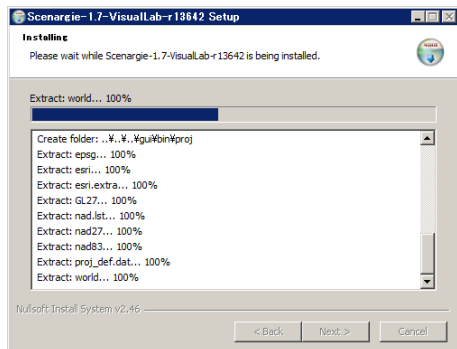


プログラムのインストール先フォルダを決定します。デフォルトのインストール先フォルダは「C:\Scenargie」です。別の場所に配置する場合には配置先のフォルダ名をキーボードから入力するか、「参照」ボタンを押して、選択画面から配置先のフォルダを選択します。

*Scenargie Visual Lab 実行中の作業用フォルダ(User Home Directory)は、インストール時には自動的にプログラムのインストール先フォルダに設定されます。

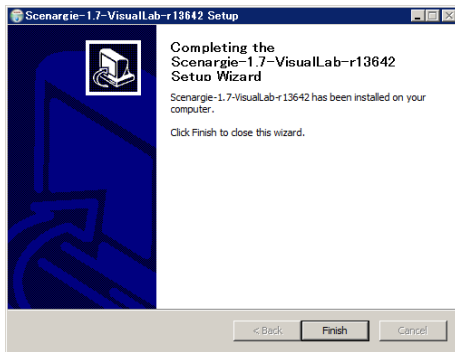
変更したい場合は、Visual Lab の「Options」メニューで変更可能です。

「Next」ボタンをクリックして、次へ進みます。



ファイルのコピーが開始されます。インストールする環境によっては時間がかかる場合がありますが、コピー作業が終了するまで待ちます。

注意！)ファイルのコピー中はセットアップを中止しないでください。



インストールの終了です。「Finish」ボタンをクリックして、セットアップウィザードを終了します。

2) Scenargie Visual Lab 起動確認



デスクトップのショートカット **Scenargie**、またはメニュー/プログラムの Scenargie より Scenargie Visual Lab が起動されることを確認します。起動時はアカウント ID 等必要情報を入力する必要があります。



以下のエラーが発生した場合、ライセンス設定を確認してください。

3) データファイルのコピー

オプションモジュールを Visual Lab と組み合わせて使用するために、データファイルが提供される場合があります。この場合、各オプションモジュールのユーザガイドを参照して Scenargie Visual Lab インストールフォルダの data/ フォルダ下にコピーしてください。

4) サンプルファイルのコピー

オプションモジュールを Visual Lab と組み合わせて使用するために、サンプルファイルが提供される場合があります。この場合、各オプションモジュールのユーザガイドを参照して Scenargie Visual Lab インストールフォルダの sample/ フォルダ下にコピーしてください。

3.1.2. Scenargie Base Simulator のインストール

ダウンロードした Scenargie Base Simulator のインストール方法について説明します。

1) パッケージの展開

Scenargie Base Simulator のパッケージを展開するとディレクトリ「scenargie_simulator」が作成され、このディレクトリ下にソースコードが展開されます。

以降の説明では、C ドライブに「C:\scenargie_simulator」が作成され展開されていることと想定します。

Scenargie Base Simulator ソースコードのディレクトリ基本構成は下記のようになります。

scenargie_simulator/2.1/

```
|-- util
|-- document
|-- package_tree
|-- scenarios_windows
|-- scenarios_linux
|-- visuallab
|-- source
    |-- base
    |-- boost
    |-- include
    |-- multisystems
    |-- objlibs
    |-- simulator
```

2) オプションモジュールの展開

Scenargie Dot Eleven Module のようなオプションモジュールは Scenargie Base Simulator ソースコードと同じディレクトリ構成となっています。従って、「scenargie_simulator」と同じ場所で展開することにより必要なファイル群が所定の位置に展開されます。オプションモジュールのインストールおよび使用方法については、オプションモジュール毎のドキュメントを参照ください。

3.1.3. Visual C++ のインストール

Scenargie Base Simulator の実行ファイルをソースコードから作成する場合、Visual C++ の開発環境が必要になります。Windows 7 32bit/64bit では Visual Studio Express 2013 for Windows Desktop (Visual C++)がサポート対象です。Windows 10 64bit では Visual Studio Express 2015 for Desktop (Visual C++)がサポート対象です。ご使用の OS に合った開発環境をインストールしてください。

Visual Studio Express 2013 for Windows Desktop (Visual C++)は下記 URL より入手可能です (2016 年 5 月現在)。

<https://www.microsoft.com/ja-jp/download/details.aspx?id=48131>

なお、本ドキュメントでは Visual Studio Express 2013 for Windows Desktop (Visual C++) を想定して説明させていただきます。適宜読み替えてご利用ください。

注意) 上記 URL では言語が選択可能ですが、本ドキュメントでは English 版の使用を前提に説明します。

3.1.4. Scenargie Base Simulator のビルド

1) コマンドプロンプトの起動

Visual C++のインストールにより作成される「Visual Studio 2013 Command Prompt」での実行を想定しています。

32bit でのビルドを行う場合は、

[スタート]-[プログラム]- [Visual Studio 2013]-[Visual Studio Tools]-[VS2013 x86 Native Tools Command Prompt]をクリックして起動される Command Prompt を使用してください。

64bit でのビルドを行う場合は、

[スタート]-[プログラム]- [Visual Studio 2013]-[Visual Studio Tools]-[VS2013 x64 Cross Tools Command Prompt]をクリックして起動される Command Prompt を使用してください。

2) ビルドの実行

Scenargie Base Simulator のビルドはビルド用ディレクトリに移動して行ってください。

注意) オプションモジュールを使用する場合のビルド方法はオプションモジュール毎のドキュメントを参照ください。

<コマンド例>

```
> cd C:\scenargie_simulator\2.1\source\base
> nmake -f makefile.win
```

正しくビルドが行われた場合、カレントディレクトリに実行ファイル「sim.exe」が作成されます。

Scenargie のサンプル makefile を使用してビルドを行う場合、BUILDTHREADS オプションを使用することでコマンドプロンプトでも並列ビルドが可能です。

BUILDTHREADS=[スレッド数]

<コマンド例>

```
> cd C:\scenargie_simulator\2.1\source\base
> nmake -f makefile.win BUILDTHREADS=4
```

注意) 一度シミュレータ実行ファイルの作成を行った後、再ビルドまたは別のシミュレータをビルドする場合は、以下のコマンドでシミュレータ実行ファイルやオブジェクトファイルを削除してください。

<コマンド例>

```
> cd C:\scenargie_simulator¥2.1¥source¥base
> nmake -f makefile.win clean
> nmake -f makefile.win
```

尚、ご使用のコンピュータに makedepend がインストールされている場合は、nmake depend コマンドが使用可能です。nmake depend を実行した場合は、以降は make コマンドのみで変更ファイルのみが再ビルドされます。

<コマンド例>

```
> cd C:\scenargie_simulator¥2.1¥source¥base
> nmake -f makefile.win depend
> nmake -f makefile.win
```

作成済みのオブジェクトファイルを全て削除して再ビルドする場合は、nmake clean 時に nmake depend の情報も削除されるため、再度 nmake depend を行ってください。

<コマンド例>

```
> cd C:\scenargie_simulator¥2.1¥source¥base
> nmake -f makefile.win clean
> nmake -f makefile.win depend
> nmake -f makefile.win
```

複数のオプションモジュールを組み合わせる場合は、マルチシステム用の makefile を使用してビルドします。

< Base Simulator に Dot Eleven Module と LTE Module を組み合わせる場合のコマンド例>

```
> cd C:\scenargie_simulator¥2.1¥source¥multisystems
> nmake -f makefile.win clean
> nmake -f makefile.win DOT11_MODULE=on LTE_MODULE=on
```

< Base Simulator に Dot Eleven Module と Multi-Agent Extension Module を組み合わせる場合のコマンド例>

```
> cd C:\scenargie_simulator\2.1\source\multisystems
> nmake -f makefile.win clean
> nmake -f makefile.win DOT11_MODULE=on MULTIAGENT_MODULE=on
```

専用 makefile とマルチシステム用 makefile を使用した場合の関係は ITS の場合以下ようになります。

```
its\makefile.win = multisystems\makefile.win ITS_MODULE=on DOT11_MODULE=on
BASE_IPV6=on
```

この他に使用可能なオプションについては、当ドキュメントのビルドオプション一覧をご参照ください。

3.1.5. Scenargie 稼働コンピュータでのライセンスの有効化

Scenargie 稼働コンピュータは、環境変数「STE_LICENSE_FILE」で指定された Scenargie 設定ファイルの情報を元にライセンスを取得します。LS の設定は、本ドキュメントの「2. LS の設定」に従って実施してください。Scenargie 稼働コンピュータでの設定は、本ドキュメントの「4. オンラインによるライセンス認証」または「5. オフラインによるライセンス認証」をご参照ください。

3.2. Linux 環境へのインストール

3.2.1. Scenargie Visual Lab のインストール

Scenargie Visual Lab の Linux 版のパッケージには以下が入っています。

- Scenargie-2.1-VisualLab-rxxx-el6-x86_64.tar.gz

1) パッケージの展開

パッケージをコマンドまたは File Browser により展開します。ディレクトリ「visuallab」が作成されます。

ディレクトリ「visuallab」を Scenargie Visual Lab インストールディレクトリと呼びます。この例ではユーザホームディレクトリに展開します。

* 以前のバージョンの Scenargie Visual Lab をアンインストールしてからインストールすることを推奨します。

<コマンド例>

```
$ cd ~
$ tar zxvf Scenargie-2.1-VisualLab-rxxx-el6-x86_64.tar.gz
$ ls visuallab/
bin data doc License plugins Scenargie
```

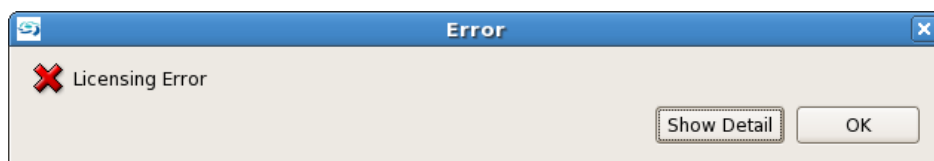
2) Scenargie Visual Lab 起動確認

Scenargie Visual Lab インストールディレクトリに移動して起動スクリプト「Scenargie」を実行し、Scenargie Visual Lab が起動されることを確認します。起動時はアカウント ID 等必要情報を入力する必要があります。

<コマンド例>

```
$ cd ~/visuallab/
$ ./Scenargie
```

以下のエラーが発生した場合、ライセンス設定を確認してください。



3) Scenargie ホームディレクトリの設定

手順 2)の起動スクリプト「Scenargie」で起動した場合、Scenargie Visual Lab インストールディレクトリが Scenargie ホームディレクトリとして使用されます。

Scenargie1.4 以前のバージョンでは Scenargie ホームディレクトリを環境変数に割り当てています。設定を解除する場合、以下のように行います。

- a) \${HOME}/.bashrc などに記述された箇所の削除

<削除箇所の例>

```
$ export SCENARGIE_HOME=${HOME}/visuallab
```

- b) 環境変数からの削除

<コマンド例>

```
$ env | grep SCENARGIE_HOME
SCENARGIE_HOME=${HOME}/visuallab
$ unset SCENARGIE_HOME
```

4) データファイルのコピー

オプションモジュールを Visual Lab と組み合わせて使用するために、データファイルが提供される場合があります。この場合、各オプションモジュールのユーザガイドを参照して Scenargie Visual Lab インストールディレクトリの data/ ディレクトリ下にコピーしてください。

5) サンプルファイルのコピー

オプションモジュールを Visual Lab と組み合わせて使用するために、サンプルファイルが提供される場合があります。この場合、各オプションモジュールのユーザガイドを参照して Scenargie Visual Lab インストールディレクトリの sample/ ディレクトリ下にコピーしてください。

注意) Linux 64bit で Video Clip の出力機能を使用する場合、使用するモジュールが 32bit 版であるため libz.so.1、libbz2.so.1 の 32bit ライブラリが必要となります。適合する 32bit ライブラリをインストールしてください。

<CentOS 6 64bit でのインストール例>

```
$ su
# yum install zlib*i686*
# yum install bzip2*i686*
```

3.2.2. Scenargie Base Simulator のインストール

ダウンロードした Scenargie Base Simulator のインストール方法について説明します。

1) パッケージの展開

Scenargie Base Simulator のパッケージを展開するとディレクトリ「scenargie_simulator」が作成され、このディレクトリ下にソースコードが展開されます。

以降の説明では、ユーザのホームディレクトリ下にパッケージが展開され、「scenargie_simulator」が作成されていることと想定します。

<コマンド例>

```
$ unzip Scenargie-2.1-BaseSimulator-rxxxx.zip
```

Scenargie Base Simulator ソースコードのディレクトリ基本構成は下記のようになります。

```
scenargie_simulator/2.1/
```

```
|-- util
|-- document
|-- package_tree
|-- scenarios_windows
|-- scenarios_linux
|-- visuallab
`-- source
    |-- base
    |-- boost
    |-- include
    |-- multisystems
    |-- objlibs
    `-- simulator
```

2) オプションモジュールの展開

Scenargie Dot Eleven Module のようなオプションモジュールは Scenargie Base Simulator ソースコードと同じディレクトリ構成となっています。従って、「scenargie_simulator」と同じ場所で展開することにより必要なファイル群が所定の位置に展開されます。

<コマンド例>

```
$ cd ~/
$ unzip Scenargie-2.1-DotElevenModule-rxxx.zip
「scenargie_simulator」というディレクトリに展開されます。
$ cd scenargie_simulator/2.1/source
$ ls
base boost dot11 include multisystems objlibs simulator
~/scenargie_simulator/2.1/source/ の下にディレクトリ「dot11」が追加されます。
```

* オプションモジュールのインストールおよび使用方法については、オプションモジュール毎のドキュメントを参照ください。

3) ビルドの実行

makefile を使用しての Scenargie Base Simulator のビルドは下記のように行います。

注意) オプションモジュールを使用する場合のビルド方法はオプションモジュール毎のドキュメントを参照ください。

<コマンド例>

```
$ cd ~/scenargie_simulator/2.1/source/base
$ make -f makefile.linux
```

正しくビルドが行われた場合、カレントディレクトリに実行ファイル「sim」が作成されます。

GNU make コマンドを使用してビルドを行う場合、-j オプションで同時に実行できるスレッド数を指定することが可能です。

```
make -j [スレッド数]
```

<コマンド例>

```
$ cd ~/scenargie_simulator/2.1/source/base
$ make -f makefile.linux -j 4
```

注意) 一度シミュレータ実行ファイルの作成を行った後、再ビルドまたは別のシミュレータをビルドする場合は、以下のコマンドでシミュレータ実行ファイルやオブジェクトファイルを削除してください。

<コマンド例>

```
$ cd ~/scenargie_simulator/2.1/source/base
$ make -f makefile.linux clean
$ make -f makefile.linux
```

尚、ご使用のコンピュータに makedepend がインストールされている場合は、make depend コマンドが使用可能です。make depend を実行した場合は、以降は make コマンドのみで変更ファイルのみが再ビルドされます。

<コマンド例>

```
$ cd ~/scenargie_simulator/2.1/source/base
$ make -f makefile.linux depend
$ make -f makefile.linux
```

作成済みのオブジェクトファイルを全て削除して再ビルドする場合は、make clean 時に make depend の情報も削除されるため、再度 make depend を行ってください。

<コマンド例>

```
$ cd ~/scenargie_simulator/2.1/source/base
$ make -f makefile.linux clean
$ make -f makefile.linux depend
$ make -f makefile.linux
```

複数のオプションモジュールを組み合わせる場合は、マルチシステム用の makefile を使用してビルドします。

< Base Simulator に Dot Eleven Module と LTE Module を組み合わせる場合のコマンド例>

```
$ cd ~/scenargie_simulator/2.1/source/multisystems
$ make -f makefile.linux clean
$ make -f makefile.linux DOT11_MODULE=on LTE_MODULE=on
```


< Base Simulator に Dot Eleven Module と Multi-Agent Extension Module を組み合わせる場合の
コマンド例>

```
$ cd ~/scenargie_simulator/2.1/source/multisystems
$ make -f makefile.linux clean
$ make -f makefile.linux DOT11_MODULE=on MULTIAGENT_MODULE=on
```

専用 makefile とマルチシステム用 makefile を使用した場合の関係は ITS の場合以下になります。

```
its/makefile.linux = multisystems/makefile.linux ITS_MODULE=on DOT11_MODULE=on
BASE_IPV6=on
```

この他に使用可能なオプションについては、当ドキュメントのビルドオプション一覧をご参照ください。

3.2.3. Scenargie 稼働コンピュータでのライセンスの有効化

Scenargie 稼働コンピュータは、環境変数「STE_LICENSE_FILE」で指定された Scenargie 設定ファイルの情報を元にライセンスを取得します。LS の設定は、本ドキュメントの「2. LS の設定」に従って実施してください。Scenargie 稼働コンピュータでの設定は、本ドキュメントの「4. オンラインによるライセンス認証」または「5. オフラインによるライセンス認証」をご参照ください。

3.3. Mac OS X 環境へのインストール

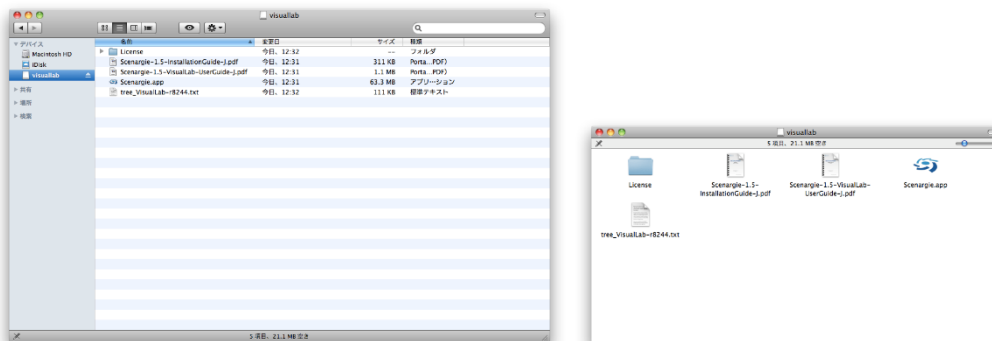
3.3.1.Scenargie Visual Lab のインストール

Scenargie Visual Lab の Mac OS X 版のパッケージは以下となります。

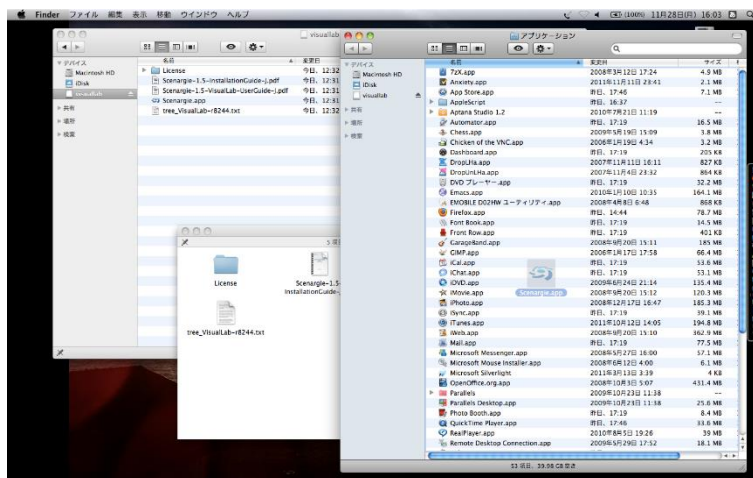
- Scenargie-2.1-VisualLab-rxxx-macosx-x86_64.dmg

1) パッケージの展開

Visual Lab のディスクイメージファイル(.dmg ファイル)をダブルクリックします。Finder のサイドバー内のデバイスとして「visualab」が認識され、以下の画面のように展開されます。



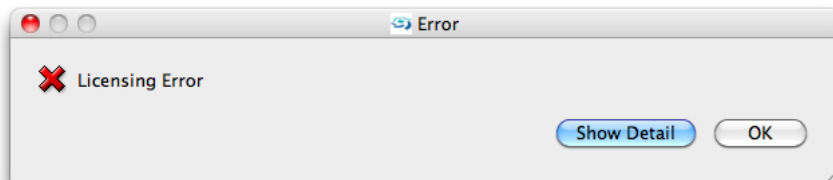
Scenargie Visual Lab のインストールは、展開されたフォルダ内の「Scenargie.app」を、Mac OS X のアプリケーションフォルダにドラッグ & ドロップでコピーすることで完了します。アプリケーションフォルダ内に、以前のバージョンの Scenargie Visual Lab が存在する場合は、アンインストールしてからインストールをしてください。



2) Scenargie Visual Lab 起動確認

コピーした Scenargie Visual Lab アプリケーションをクリックして起動を確認します。起動時はアカウント ID 等必要情報を入力する必要があります。

以下のエラーが発生した場合、ライセンス設定を確認してください。



3) データファイルのコピー

オプションモジュールを Visual Lab と組み合わせて使用するために、データファイルが提供される場合があります。この場合、各オプションモジュールのユーザガイドを参照して Scenargie Visual Lab インストールディレクトリの data/ ディレクトリ下にコピーしてください。

/Applications/Scenargie.app/

-- Contents

```

|-- Frameworks
|-- MacOS
|-- Resources
|-- bin
|   |-- proj
|-- data
|-- plugins
|   |-- imageformats
|   |-- sqldrivers
-- sample

```

4) サンプルファイルのコピー

オプションモジュールを Visual Lab と組み合わせて使用するために、サンプルファイルが提供される場合があります。この場合、各オプションモジュールのユーザガイドを参照して Scenargie Visual Lab インストールディレクトリの sample/ ディレクトリ下にコピーしてください。

/Applications/Scenargie.app/

```

|-- Contents
|   |-- Frameworks
|   |-- MacOS
|   |-- Resources
|   |-- bin
|   |   |-- proj
|   |-- data
|   |-- plugins
|   |   |-- imageformats
|   |   |-- sqldrivers
|-- sample

```

3.3.2. Scenargie Base Simulator のインストール

ダウンロードした Scenargie Base Simulator のインストール方法について説明します。

1) パッケージの展開

Scenargie Base Simulator のパッケージを展開するとディレクトリ「scenargie_simulator」が作成され、このディレクトリ下にソースコードが展開されます。

<コマンド例>

```
$ unzip Scenargie-2.1-BaseSimulator-rxxxx.zip
```

Scenargie Base Simulator ソースコードのディレクトリ基本構成は下記のようになります。

```

scenargie_simulator/2.1/
|-- util
|-- document
|-- package_tree
|-- scenarios_windows
|-- scenarios_linux
|-- visuallab
|-- source
|   |-- base
|   |-- boost
|   |-- include
|   |-- multisystems
|   |-- objlibs

```

```
`-- simulator
```

以降の説明では、ユーザのホームディレクトリ下に「scenargie_simulator」が作成され展開されていることと想定します。

2) ビルドの実行

makefile を使用しての Scenargie Base Simulator のビルド確認は下記のように行います。

注意) オプションモジュールを使用する場合のビルド方法はオプションモジュール毎のドキュメントを参照ください。

<コマンド例>

```
$ cd ~/scenargie_simulator/2.1/source/base
$ make -f makefile.linux
```

正しくビルドが行われた場合、カレントディレクトリに実行ファイル「sim」が作成されます。

GNU make コマンドを使用してビルドを行う場合、-j オプションで同時に実行できるスレッド数を指定することが可能です。

make -j [スレッド数]

<コマンド例>

```
$ cd ~/scenargie_simulator/2.1/source/base
$ make -f makefile.linux -j 4
```

注意) 一度シミュレータ実行ファイルの作成を行った後、再ビルドまたは別のシミュレータをビルドする場合は、以下のコマンドでシミュレータ実行ファイルやオブジェクトファイルを削除してください。

<コマンド例>

```
$ cd ~/scenargie_simulator/2.1/source/base
$ make -f makefile.linux clean
$ make -f makefile.linux
```

尚、ご使用のコンピュータに makedepend がインストールされている場合は、make depend コマンドが使用可能です。make depend を実行した場合は、以降は make コマンドのみで変更ファイルのみが再ビルドされます。

<コマンド例>

```
$ cd ~/scenargie_simulator/2.1/source/base
$ make -f makefile.linux depend
$ make -f makefile.linux
```

作成済みのオブジェクトファイルを全て削除して再ビルドする場合は、make clean 時に make depend の情報も削除されるため、再度 make depend を行ってください。

<コマンド例>

```
$ cd ~/scenargie_simulator/2.1/source/base
$ make -f makefile.linux clean
$ make -f makefile.linux depend
$ make -f makefile.linux
```

3) オプションモジュールの展開

Scenargie Dot Eleven Module のようなオプションモジュールは Scenargie Base Simulator ソースコードと同じディレクトリ構成となっています。従って、「scenargie_simulator」と同じ場所で展開することにより必要なファイル群が所定の位置に展開されます。オプションモジュールのインストールおよび使用方法については、オプションモジュール毎のドキュメントを参照ください。

<コマンド例>

```
$ cd ~/
$ unzip Scenargie-2.1-DotElevenModule-rxxx.zip
「scenargie_simulator」というディレクトリに展開されます。
$ cd scenargie_simulator/2.1/source
$ ls
base boost dot11 include multisystems objlibs simulator
~/scenargie_simulator/2.1/source/ の下にディレクトリ「dot11」が追加されます。
```

複数のオプションモジュールを組み合わせる場合は、source/multisystems 以下に含まれるマルチシステム用の makefile を使用してビルドします。

< Base Simulator に Dot Eleven Module と LTE Module を組み合わせる場合のコマンド例>

```
$ cd ~/scenargie_simulator/2.1/source/multisystems
$ make -f makefile.linux clean
$ make -f makefile.linux DOT11_MODULE=on LTE_MODULE=on
```

< Base Simulator に Dot Eleven Module と Multi-Agent Extension Module を組み合わせる場合のコマンド例>

```
$ cd ~/scenargie_simulator/2.1/source/multisystems
$ make -f makefile.linux clean
$ make -f makefile.linux DOT11_MODULE=on MULTIAGENT_MODULE=on
```

専用 makefile とマルチシステム用 makefile を使用した場合の関係は ITS の場合以下ようになります。

```
its/makefile.linux = multisystems/makefile.linux ITS_MODULE=on DOT11_MODULE=on
BASE_IPV6=on
```

この他に使用可能なオプションについては、当ドキュメントのビルドオプション一覧をご参照ください。

3.3.3. Scenargie 稼働コンピュータでのライセンスの有効化

Scenargie 稼働コンピュータは、環境変数「STE_LICENSE_FILE」で指定された Scenargie 設定ファイルの情報を元にライセンスを取得します。LS の設定は、本ドキュメントの「2. LS の設定」に従って実施してください。Scenargie 稼働コンピュータでの設定は、本ドキュメントの「4. オンラインによるライセンス認証」または「5. オフラインによるライセンス認証」をご参照ください。

3.4. ビルドオプション一覧

3.4.1. 全モジュール共通のビルドオプション

以下のビルドオプションは、すべてのモジュールにて使用可能です。

BASE_IPV6=on	IPv6 を使用します。
BASE_ISCDHCP=on	ISC-DHCP の機能を有効にします。 ※Linux のみ有効です。
BASE_LIBPCAP=on	TraceBasedApp アプリケーションで利用する Pcap 形式のデータの使用を有効にします。
FUPM=on	FUPM の機能を有効にします。 ※別途 Fast Urban Propagation Module が必要になります。
HFPM=on	HFPM の機能を有効にします。 ※別途 High Fidelity Propagation Module が必要になります。

3.4.2. Multisystems で使用可能なビルドオプション

以下のビルドオプションは、multisystems の makefile を使用したビルド時にのみ使用可能です。それぞれ対応するオプションモジュールが必要になります。

DOT11_MODULE=on	Dot Eleven Module の機能を有効にします。
DOT11_ADVANCED_MODULE=on	Dot Eleven Module Advanced の機能を有効にします。
ITS_MODULE=on	ITS Extension Module の機能を有効にします。
LTE_MODULE=on	LTE Extension Module の機能を有効にします。
MULTIAGENT_MODULE=on	Multi-Agent Extension Module の機能を有効にします。

4. オンラインによるライセンス認証

Scenargie 稼働コンピュータが LS とネットワークで接続されている場合は LS から、Scenargie 稼働コンピュータがインターネットに接続されている場合は SS から、ネットワーク経由でライセンス認証を行うことができます(オンラインチェックアウト)。

4.1. オンラインチェックアウトの手順

オンラインチェックアウトを行うには Scenargie 設定ファイル(scenargie_setting.txt)の作成が必要になります。

Scenargie 設定ファイルは、各行ごとに「<タグ名>=<値>」の書式で記載します。オンラインチェックアウトでは、Scenargie 設定ファイルに LS 稼働コンピュータの IP アドレス、アカウント ID、メールアドレスを指定し、環境変数 STE_LICENSE_FILE に Scenargie 設定ファイルのパスを指定します。

<Scenargie 設定ファイルの例>

```
license-server=192.168.0.1
account-id=012345
email=aaa@bbb.ccc
```

Scenargie 設定ファイルでは、オンラインチェックアウト用のタグとして以下のタグが使用可能です。

- license-server
LS からのオンラインチェックアウトを行う場合は LS 稼働コンピュータの IP アドレス、SS からのオンラインチェックアウトを行う場合は ss.spacetime-eng.com を指定します。
- account-id
アカウント ID(6 桁の数字)を指定します。アカウント ID は STE ウェブサイトの Dashboard、My accounts より確認できます。不明な場合は、アカウントの管理者にお問い合わせください。
- email
メールアドレスを指定します。

4.1.1. 環境変数の設定

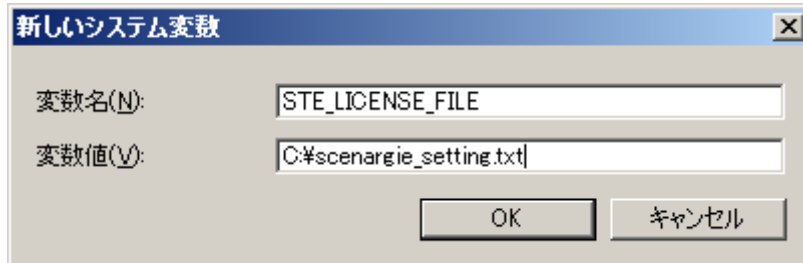
環境変数 STE_LICENSE_FILE は以下の方法で設定します。

4.1.1.1. Windows での設定方法

マイコンピュータを右クリックし、プロパティをクリックします。

「システムの詳細設定」をクリックし、「環境変数」ボタンを押して以下のように設定します。

この例では Scenargie 設定ファイルを C:\\$scenargie_setting.txt に配置しています。



4.1.1.2. Linux / Mac OS X での設定方法

Scenargie 設定ファイルのパスを指定します。

この例では Scenargie 設定ファイルをユーザディレクトリ直下に配置しています。

<設定例>

```
export STE_LICENSE_FILE=/home/user/scenargie_setting.txt
```

シェルに bash をご使用の場合は、上記設定を\${HOME}/.bashrc に記述することにより、端末起動時に環境変数が自動的に設定されます。

Mac OS X で GUI より Visual Lab を起動する場合は、以下の手順で環境変数を設定してください。この手順には管理者権限が必要になります。

- 1) /etc/launchd.conf を作成します。
- 2) 以下の内容を/etc/launchd.conf に記載します。

```
setenv STE_LICENSE_FILE /Users/user/scenargie_setting.txt
```
- 3) launchctl にて設定を有効化します。

```
launchctl < /etc/launchd.conf
```
- 4) コンピュータを再起動します。

管理者権限がない場合は、以下のように環境変数を設定してください。尚、この設定は OS の再起動時に無効になりますので、再起動時は再設定してください。

<設定例>

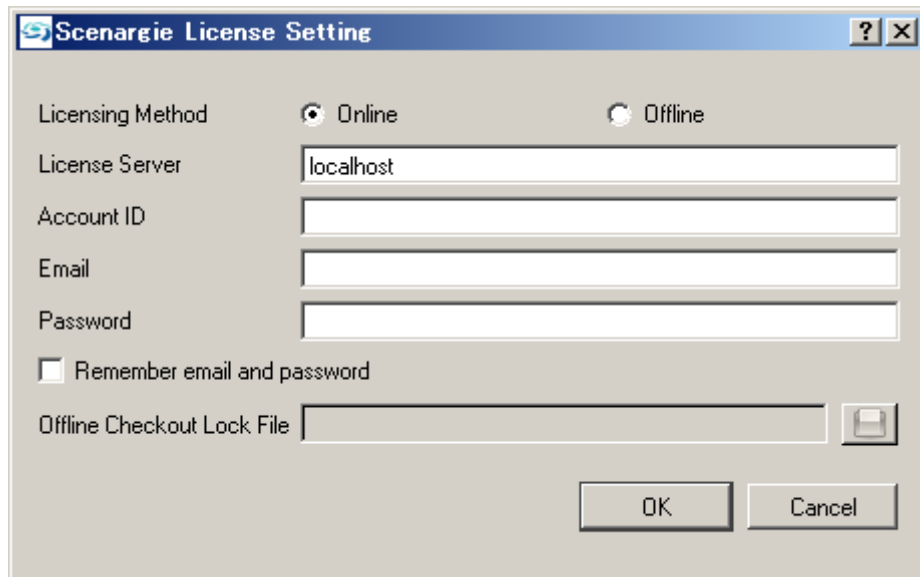
```
launchctl setenv STE_LICENSE_FILE /Users/user/scenargie_setting.txt
```

4.1.2. ユーザ認証

Scenargie を実行する際に、ユーザ認証およびライセンス認証が行われます。ユーザ認証に必要な情報が不足している場合、シミュレーション実行時にユーザ情報を設定する必要があります。

- Visual Lab で実行する場合

Visual Lab 起動時に、ユーザ認証設定画面が表示されます。



The image shows a Windows-style dialog box titled "Scenargie License Setting". It contains the following fields and controls:

- Licensing Method:** Two radio buttons, "Online" (selected) and "Offline".
- License Server:** A text input field containing "localhost".
- Account ID:** An empty text input field.
- Email:** An empty text input field.
- Password:** An empty text input field.
- Remember email and password:** An unchecked checkbox.
- Offline Checkout Lock File:** A text input field with a file icon button to its right.
- Buttons:** "OK" and "Cancel" buttons at the bottom right.

オンラインチェックアウトを行う場合は Licensing Method の Online にチェックを入れ、License Server(LS の IP アドレスまたは ss.spacetime-eng.com)、Account ID、Email、Password を入力して OK します。

注意) 「Remember email and password」にチェックを入れた場合、パスワードは暗号化されて保存されます。

Visual Lab が起動すればユーザ認証及びライセンス認証に成功しています。

- コマンドラインで実行する場合

ご使用のプラットフォームの「Scenargie Base Simulator のインストール」に従ってシミュレータをビルドし、適当なシナリオでシミュレーションを実行します。

メールアドレスまたはパスワードが Scenargie 設定ファイルに記載されていない場合、メールアドレス及びパスワードを聞かれますので入力します。

シミュレーションが実行されればユーザ認証およびライセンス認証に成功しています。

尚、環境変数 STE_LICENSE_FILE を設定せずにユーザ認証を行った場合、コマンドラインでの実行ではコマンド実行時のカレントディレクトリに、Visual Lab での実行では Windows、Linux の場合は Visual Lab のインストールディレクトリに、Mac では Application ディレクトリに自動的に Scenargie 設定ファイルが作成されます。

注意) "Would you like to save the password for "mail address"?"でパスワードを保存した場合、パスワードは暗号化された状態で Scenargie 設定ファイルに保存されます。

5. オフラインによるライセンス認証

オフラインによるライセンス認証(オフラインチェックアウト)は、LS または SS から一時的にライセンスを取りはずし、Scenargie 稼働コンピュータにノード固定のライセンスとして登録する機能です。オフラインチェックアウトを行うと、LS 及び SS のとのやり取りが不要になります。尚、オフラインチェックアウトしている数だけ、オンラインチェックアウトできるライセンス数は減少します。

オフラインチェックアウト時は、オフラインチェックアウトを行っているユーザのメールアドレスが Moved/Checked out by(SS)、Checked out by(LS)に表示されます。

注意)オフラインチェックアウトが行える期間は最大で 7 日間となります。オフラインチェックアウト後のキャンセル(オフラインチェックアウト期間の短縮)はできません。また、シミュレーションはオンラインチェックアウトとオフラインチェックアウトのどちらかのみを利用して実行されるため、混合して使用することはできません。

5.1. オフラインチェックアウトの操作手順(Scenargie 稼働コンピュータでの操作その 1)

- 1) Base Simulator のパッケージを展開し、展開したパッケージの scenargie_simulator/2.1/util に移動します。
- 2) Scenargie 稼働コンピュータのアーキテクチャに合ったディレクトリに移動し、make-coreq-file(.exe)を実行します。
- 3) ユーザ情報(アカウント ID、メールアドレス、パスワード)を入力します。尚、Scenargie 設定ファイルに各ユーザ情報が記載されている場合は、Scenargie 設定ファイルに記載されているユーザ情報を自動的に使用します。

Scenargie 設定ファイルにユーザ情報が記載されている状態で、別のユーザ情報を使用してオフラインチェックアウトリクエストファイルを作成したい場合には、--new オプションを使用することでコマンドラインから入力したユーザ情報を使用してオフラインチェックアウトリクエストファイルを作成することが可能です。


<コマンド例>

```
$ ./make-coreq-file --new
$
```


注意) Red Hat Enterprise Linux 7 及び CentOS 7 では make-coreq-file の実行に root 権限が必要です。

5.2. オフラインチェックアウトの操作手順 (LS または SS での操作)

5.2.1. LS からオフラインチェックアウトする場合

- 1) offline_co_request.dat ファイルを Scenargie 稼働コンピュータから LS 稼働コンピュータに移動し、LS 稼働コンピュータでウェブブラウザより <http://localhost:8082> にアクセスします。
- 2) License management をクリックします。
- 3) オフラインチェックアウトをするライセンスを選択し、 の「Offline checkout selected」をクリックします。
- 4) Checkout expiration にオフラインチェックアウトを行う期間を指定し、Checkout request file に Scenargie 稼働コンピュータで作成したオフラインチェックアウトリクエストファイル (offline_co_request.dat) を指定して Offline checkout from LS をクリックします。
- 5) Confirmation を確認し、OK をクリックします。
- 6) オフラインチェックアウトロックファイル (offline_co_lock.dat) がダウンロードされます。
- 7) ダウンロードしたオフラインチェックアウトロックファイルを Scenargie 稼働コンピュータに移動します。

5.2.2. SS からオフラインチェックアウトする場合

- 1) ウェブブラウザより STE のウェブサイト (<https://www.spacetime-eng.com>) にアクセスします。
- 2) 会員ページからログインし、「License management」に移動します。
- 3) オフラインチェックアウトをするライセンスを選択し、 の「Offline checkout」をクリックします。
- 4) Checkout expiration にオフラインチェックアウトを行う期間を指定し、Checkout request file に Scenargie 稼働コンピュータで作成したオフラインチェックアウトリクエストファイル (offline_co_request.dat) を指定して Offline checkout from SS をクリックします。
- 5) Confirmation を確認し、OK をクリックします。
- 6) オフラインチェックアウトロックファイル (offline_co_lock_*.dat) がダウンロードされます。

5.3. オフラインチェックアウトの操作手順 (Scenargie 稼働コンピュータでの操作その 2)

ここでは、Windows でオフラインチェックアウトロックファイルを C ドライブ直下 (C:\offline_co_lock.dat) に配置したものと想定します。

Scenargie 稼働コンピュータで、Scenargie 設定ファイル (scenargie_setting.txt) の「offline-checkout-lock-file」タグにオフラインチェックアウトロックファイルのパスを設定します。

<Scenargie 設定ファイル記入例>

```
offline-checkout-lock-file=C:¥offline_co_lock.dat
```

オフラインチェックアウト用のタグとして以下が使用可能です。

- offline-checkout-lock-file<Num>
オフラインチェックアウトロックファイルへのパスを指定します。
当タグは複数使用可能です。

<Scenargie 設定ファイル記入例>

```
offline-checkout-lock-file1=C:¥offline_co_lock1.dat  
offline-checkout-lock-file2=C:¥offline_co_lock2.dat
```

5.3.1.環境変数の設定

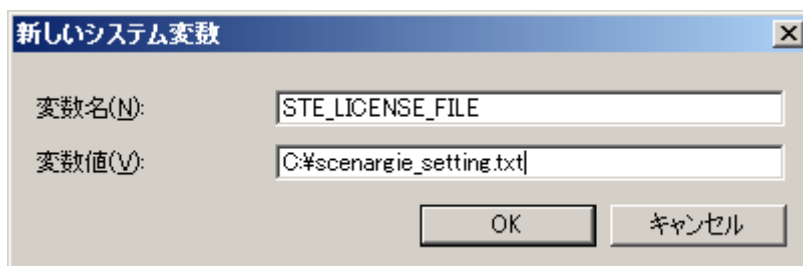
環境変数 STE_LICENSE_FILE は以下の方法で設定します。

5.3.1.1. Windows での設定方法

マイコンピュータを右クリックし、プロパティをクリックします。

「システムの詳細設定」をクリックし、「環境変数」ボタンを押して以下のように設定します。

この例では Scenargie 設定ファイルを C:¥scenargie_setting.txt に配置しています。



5.3.1.2. Linux / Mac OS X での設定方法

Scenargie 設定ファイルのパスを指定します。

この例では Scenargie 設定ファイルをユーザディレクトリ直下に配置しています。

<設定例>

```
export STE_LICENSE_FILE=/home/user/scenargie_setting.txt
```

シェルに bash をご使用の場合は、上記設定を\${HOME}/.bashrc に記述することにより、端末起動時に環境変数が自動的に設定されます。

Mac OS X で GUI より Visual Lab を起動する場合は、以下の手順で環境変数を設定してください。

- 1) /etc/launchd.conf を作成します。
- 2) 以下の内容を/etc/launchd.conf に記載します。
setenv STE_LICENSE_FILE /Users/user/scenargie_setting.txt
- 3) launchctl にて設定を有効化します。
launchctl < etc/launchd.conf
- 4) コンピュータを再起動します。

管理者権限がない場合は、以下のように環境変数を設定してください。尚、この設定は OS の再起動時に無効になりますので、再起動時は再設定してください。

<設定例>

```
launchctl setenv STE_LICENSE_FILE /Users/user/scenargie_setting.txt
```

注意) Red Hat Enterprise Linux 7 及び CentOS 7 ではオフラインチェックアウトでシミュレータを実行する際には root 権限が必要です。

