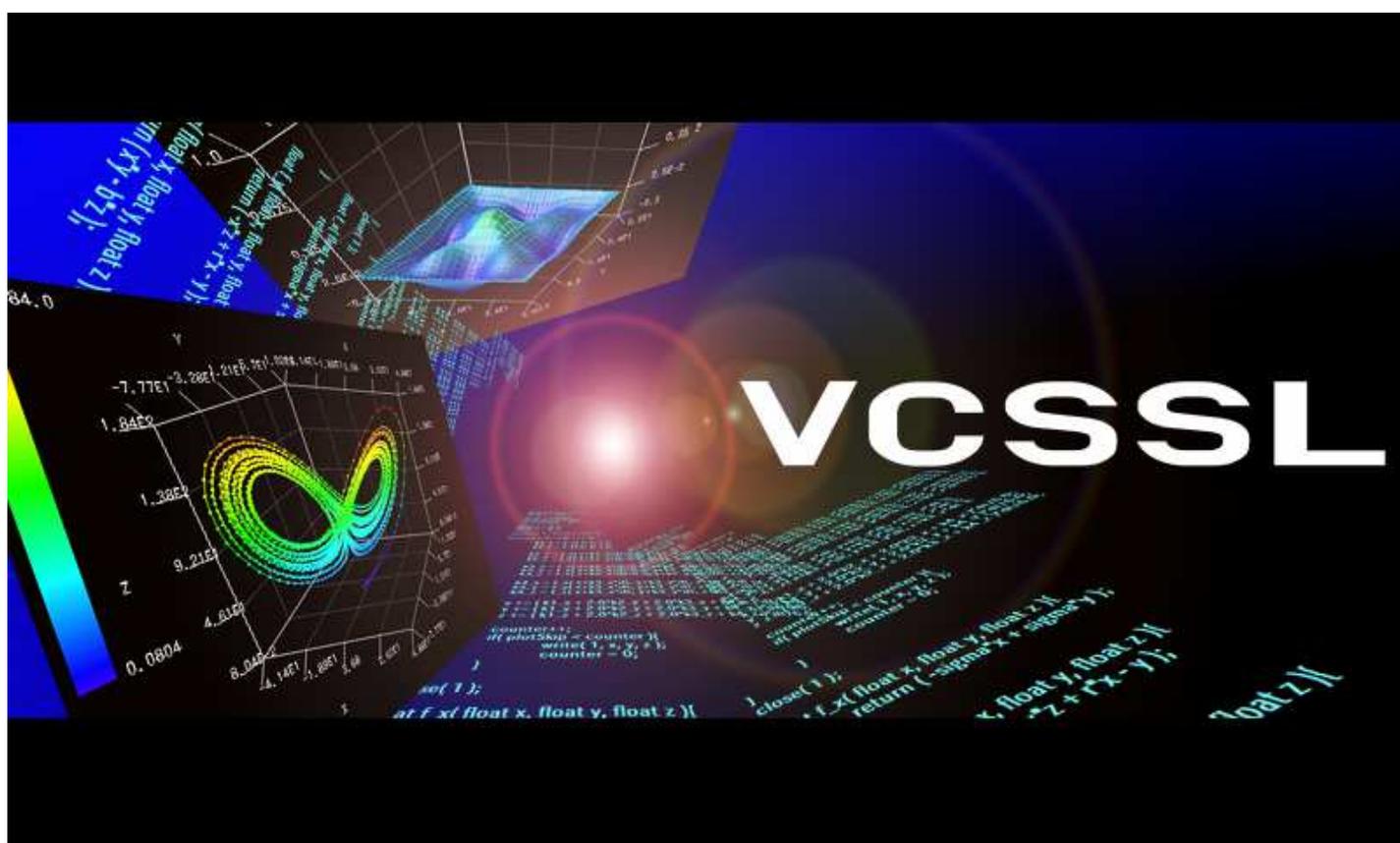


VCSSL オフィシャル リファレンス ガイド

Start UP

- スタートアップガイド -



第 9 版

はじめに



この「 VCSSL スタートアップガイド 」は、
プログラミング言語 VCSSL の紹介や、
導入のサポートを行うためのガイドです。

■ それでは、次ページから

プログラミング言語 VCSSL について見ていきましょう !



▶▶ NEXT



▶▶ NEXT

■ 2011 年誕生の新しいプログラミング言語



VCSSL は、2011 年誕生の新しいプログラミング言語です。

最初のバージョンである VCSSL1.0 は、2011 年の年始めに正式版の公開がスタートしました。現行バージョンである VCSSL2.x 系は、VCSSL1.0 を拡張してグラフィックス制御などの諸機能を強化したものであり、2011 年の 9 月に正式版の公開を迎えました。

■ スクリプト言語だから、書いてすぐ実行

VCSSL のプログラムを開発するのに、専用のソフトなどを立ち上げる必要は一切ありません。

例えば、メモ用のソフトなどでさっと数行書き、「.vcssl」の拡張子を付けて保存すれば、それだけで VCSSL プログラムの完成です。あとは、それをマウスでダブルクリックするだけで、自動的に実行できます（※詳しくは後の「VCSSL をはじめよう！」をご覧ください）。

■ 実行環境ごと持ち運べて、どこでも動作する

VCSSL の実行環境（ランタイム）は、様々なオペレーティングシステムに対応し、インストール不要で動作します。例えば、USB フラッシュメモリーなどに VCSSL の実行環境とプログラムを入れておき、持ち運びながら外出先のコンピューターで使用する事なども可能です。その際、面倒な機種依存性などに悩まされる心配は要りません。

つまり VCSSL プログラムは、どこでも開発できて、実行環境ごと持ち運べて、そしてどこでも動作するのです。

VCSSLの3つの特徴



■ 自作のプログラムに、実行環境を同梱して配布できる

上で述べた特徴に加えて、VCSSL にはさらに便利な特徴があります。それは、自作のプログラムに、VCSSL の実行環境(ランタイム)を同梱して配布できるという事です。この事は、VCSSL の実行環境に関するライセンス(使用許諾)でも、公式に認められています(※詳しくは後の「 VCSSL をはじめよう! 」をご覧ください)。

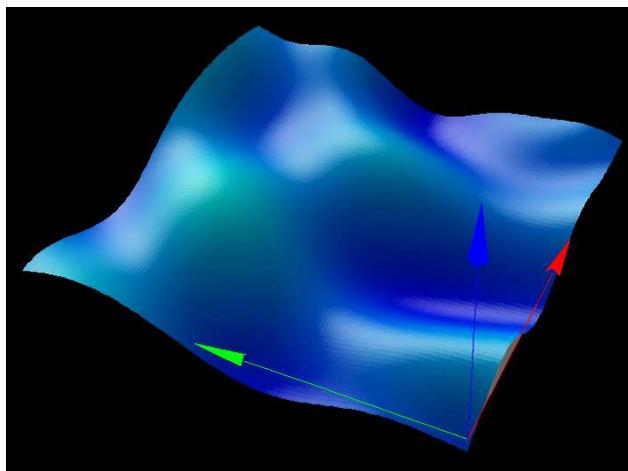
VCSSL の実行環境は、先に述べたように、様々なオペレーティングシステムにおいて、インストール不要で動作します。つまり、自作のプログラムに実行環境を同梱して配布すると、どこの誰でも、そのプログラムを入手してすぐに使う事が可能です。

自分で開発したソフトやゲームを、機種や動作環境などに縛られず、できるだけ多くの人に、できるだけ手軽に使ってもらいたいといった場合、VCSSL は最適な選択肢の一つであると言えるでしょう。

■ 日常計算から 3 次元コンピューターグラフィックスまでをカバー

VCSSL の機能拡張はまだ進行中であり、今後もさらなる多機能化が進むでしょう。

しかし、現時点の水準でも、それなりに幅広い層の要求に応える事が可能となっています。例を挙げるなら、日常のちょっとした計算用途はもちろん、科学技術計算や、GUI を用いたアプリケーション開発、ひいてはリアルタイムの 2 次元 / 3 次元コンピューターグラフィックスまでの広範囲をカバーできます。



■ C 言語系のシンプルな文法

VCSSL は、C 言語に似た文法を持っています。そして、C 言語系の言語の中でもかなりシンプルで、簡潔な記述が可能です。すでに C 言語を扱えるユーザーなら、おそらく公式ガイドを流し読みしただけで、基本的な部分はすぐに習得できるでしょう。

もちろん、VCSSL はプログラミング未経験のユーザーにもおすすめです。文法がシンプルで入門しやすい事に加えて、他言語へのステップアップの足がかりにもなるからです。C 言語系の文法を持つ言語は実に様々な種類があり、どれも第一線で活躍しています。

```
int a = 1 ;  
int b = 2 ;  
int c = a + b ;  
print( c ) ;
```

■ 公式 WEB サイトには、無償の各種ガイドが充実

VCSSL の公式サイトでは、VCSSL の各種公式ガイドを WEB 版 / PDF 版ともに無償で公開しています。公式ガイドでは、「 Hello, World 」の画面表示からリアルタイム 3 次元コンピューターグラフィックスまで、VCSSL プログラミングのすべてを細かく解説しています。サンプル類も豊富で、これだけで VCSSL の入門から実践活用までを強力にサポートします。

その他にも、VCSSL 公式 WEB サイトでは、様々な VCSSL 関連情報を発信しています。ぜひ一度ご訪問ください。

<http://www.rinearn.com/vcssl/>

▲ VCSSL 公式 WEB サイト

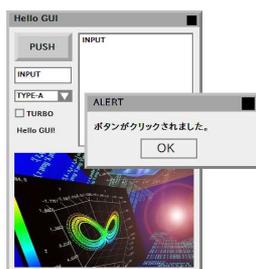


■ 日常のちょっとした計算をすぐにできる

例えば、日常のあるとき、ちょっとした計算が必要になったとしましょう。電卓ソフトを起動して手で計算してもいいですが、似たような計算を、値を変えて何度も行うような場合には面倒です。

そんな時は、メモ用のソフトなどを開き、計算内容をさっと書いて、拡張子「.vcssl」をつけて保存しましょう。あとはそれをダブルクリックするだけで、すぐに計算を実行できます（※詳しくは後の「VCSSLをはじめよう！」をご覧ください）。

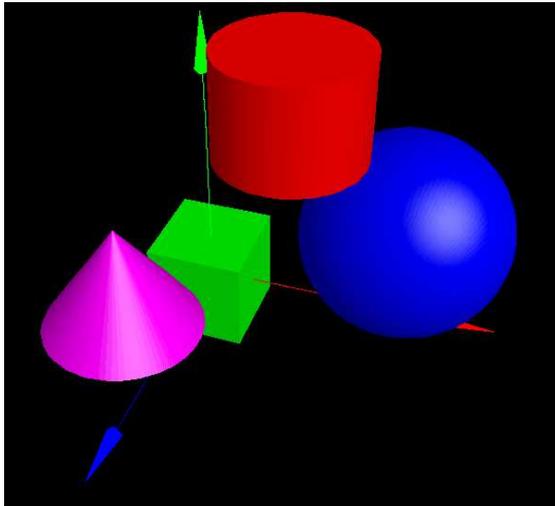
■ 簡単に GUI を用いたアプリケーションを開発できる



VCSSLでは、ウィンドウやボタンなどの GUI 部品を備えたアプリケーションを、簡単に開発する事ができます。

例えばウィドウの生成はたった 1 行で済み、それにボタンを生成して配置するのもたった 2 行でできます。ボタンなどがクリックされた際の処理も、あらかじめ決まった名前の関数内に書くだけでよく、その処理が適切なタイミングで自動的に呼ばれます。

※ 上の画像は模式図です。実際の外観は、オペレーティングシステムの種類などによって異なります。



■ 2次元 / 3次元コンピューターグラフィックスを扱える

VCSSL は、言語仕様として 2次元 / 3次元コンピューターグラフィックスを標準サポートしています。共にリアルタイムでの処理に対応しており、アニメーション描画などを行う事が可能となっています。

VCSSLの3次元コンピューターグラフィックス機能は、ソフトウェア実装のレンダラー（描画エンジン）

で処理されるため、グラフィックスボードなどの特別なハードウェアの有無に関わらず、どのようなコンピューター上においても動作します。その分、ハードウェアを用いる場合と比較してパフォーマンスは制限されますが、それでも数十万ポリゴン/秒の性能水準を実現しています。

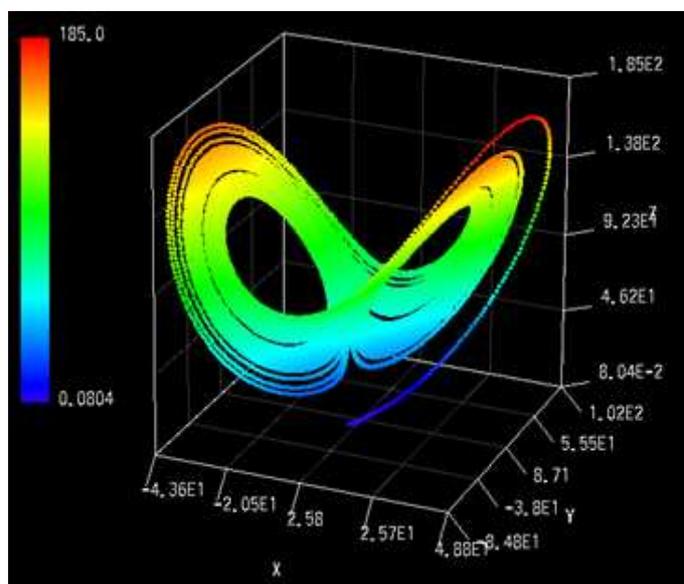
■ 科学技術系のシミュレーションも

VCSSL を用いて、科学技術系のシミュレーションや数値解析を行う事もできます。

この用途では、演算性能の高低が非常に重要なファクターとなりますが、2011 年秋現在の最新の VCSSL エンジン（ VCSSL プログラムを処理するインタプリタ ）は大体 100MFLOPS 程度のパフォーマンスを発揮します。これは具体的には、1 秒間に約 1 億回の 64bit 浮動小数点演算を行える性能です。このパフォーマンスは、そう重くない数値積分や力学シミュレーションなら、ある程度現実的に扱える水準です。

また、VCSSLは拡張APIライブラリで2次元/3次元グラフィックソフトの制御をサポートしており、プログラムの中だけで、計算から結果のグラフプロットまでを完結させる事ができます。グラフィックソフトと通信し続ける事で、計算内容をリアルタイムにアニメーションで可視化する事も可能です。

さらに、VCSSLではGUIを簡単に活用する事ができるので、パラメータ変数をスライダーバーの操作によりワンタッチで変更できるようなシミュレーションプログラムも、比較的簡単に作成する事が可能です。





■ プログラムの開発

ここからは、実際に VCSSL でプログラミングを行ってみましょう。

それでは、早速 VCSSL のプログラムを開発します。まず、メモ用のものでも何でもよいので、何かテキストを書くソフトを起動してください。そして、以下のように記述してみてください。

```
print ("Hello, World !");
```

記述が完了したら、「**Test.vcssl**」という名前を付けて適当な場所に保存してください。なお、ファイルの種類を選べる場合は、「**すべてのファイル**」などを選択する必要があります。以上で、最初のプログラムは完成です。



このプログラムは、画面に

「**Hello, World !**」

というメッセージを表示させるものになっています。この Hello, World ! というメッセージは、様々なプログラミング言語において最初に表示させる事が一種の伝統となっている、非常に有名なメッセージです。

■ 実行環境の準備

■ VCSSL ランタイムのダウンロード

VCSSL のプログラムを実行するには、「VCSSL ランタイム」を入手する必要があります。VCSSL ランタイムとは、VCSSL のプログラムを動作させる、一種のプレイヤーのようなソフトウェアです。下記 URL から無償で入手できますので、適当な場所にダウンロードしてください。



ダウンロードしたファイルは ZIP 形式で圧縮されていますので、解凍してください（ファイルを右クリックして「すべて展開」などを選択すると解凍できます）。インストール作業などは不要なので、解凍したフォルダは、どこか分かりやすい適当な場所に置いておくだけで OK です。

■ Java のアップデート

VCSSL ランタイムを起動する前に、下記の Java 公式サイトにアクセスし、Java を最新版にアップデートしておく事をおすすめします。

<http://java.com/ja/>

※ VCSSL ランタイムが正常に動作しない、または起動できない場合は、古い Java 実行環境をすべてアンインストールしてから、最新版をインストールしなおす事をおすすめします。

※ Oracle と Java は、Oracle Corporation 及びその子会社、関連会社の米国及びその他の国における登録商標です。文中の社名、商品名等は各社の商標または登録商標である場合があります。

■ 実行環境をインストールせずに実行する

VCSSL ランタイムは、インストールせずにそのまま利用する事が可能です。これは、プログラムとランタイムを USB フラッシュメモリーなどで持ち運び、複数のコンピューターで利用したい場合などに大変便利です。ここでは、とりあえずインストールせずに利用してみましょう。

なお、VCSSL ランタイムをインストールし、より便利に利用する事も可能です。その方法については後の「**実行環境のインストール**」をご参照ください。

・ダブルクリックで起動する

VCSSL ランタイムをインストールせずに起動するには、解凍した VCSSL ランタイムのフォルダ内の、「bin」というフォルダ内にある、「VCSSL.jar」という名前のファイルをダブルクリックしてください。

```
vcssl2_*_* フォルダ
- bin フォルダ
  - VCSSL.jar ( JAR ファイル )
```

すると、ファイルを選択するウィンドウが表示されるので、実行したい VCSSL プログラム(拡張子が.vcsslである事が必要)を選択してください。これでプログラムが実行されます。

・コマンド入力端末から起動する

ダブルクリックで上手く実行できない場合は、マウスで右クリックして、「Java Runtime で開く」などのメニューがあれば選択してください。

どうしても実行できない場合には、コマンド入力端末を起動し、VCSSL.jarのある場所まで移動した後、以下のようにコマンド入力して起動する事も可能です。

```
java -Xmx128m -jar VCSSL.jar
```

この例では 128MB のメモリー容量を確保して起動します。メモリー容量は-Xmx***mの部分にメガバイト単位で指定してください。

■ VCSSL の基本画面 (VCSSL コンソール)

「 Test.vcssl 」を起動すると、画面に黒いウィンドウが表示され、白い文字で「 Hello, World ! 」と表示されます。



※画像は模式図です。実際の外観は、オペレーションシステムの種類などによって異なります。

この黒い画面は「 VCSSL コンソール 」というもので、VCSSL プログラムの実行中に出現し、ここにメッセージなどを表示する事ができます。VCSSL コンソールを閉じると、VCSSL プログラムの実行も終了します。

```
-----
エラー 1
[ 発生箇所 ] プログラム「 UNKNOWN_PROGRAM 」の 3 行目
[ 処理内容 ] r_x = 2.0
[ 詳細情報 ]
               宣言されていない変数「r_x」を呼び出しています。
-----
エラー 2
[ 発生箇所 ] プログラム「 UNKNOWN_PROGRAM 」の 11 行目
[ 処理内容 ] f = fun(x)
[ 詳細情報 ]
               関数「 fun 」
               を呼び出しましたが、引数の型/個数が適合しませんでした。
               以下に引数の候補を表示します：

               fun( int[] a, int[] b )
               fun( int a, int b )
```

VCSSL コンソールには、自分でメッセージを表示するだけでなく、システム側からメッセージが表示される事もあります。

それはプログラム中に記述ミスがあった場合や、実行中にエラーが発生した場合などで、その内容に関連したシステムメッセージが表示されます。

システムメッセージを表示している際には、**VCSSL コンソールの色が青色**になります。

■ 実行環境のインストール(任意)

※この作業は任意なので、行わない場合は次へお進みください。

VCSSL プログラムを日常的に利用するようなコンピューターでは、VCSSL ランタイムをインストールしておくとは便利です。

インストール作業を行うと、VCSSL プログラムをダブルクリックするだけで自動的に実行したり、コマンド入力端末のどこからでも VCSSL プログラムを実行したりする事が可能になります。

■ インストール作業 — Windows の場合

Windows をご使用の場合は、インストール作業は非常に簡単です。具体的には、下記のように拡張子の関連付けを行うだけで済みます。

まず、先ほど作成した「 Test.vcssl 」を右クリックし、メニューから「 **プログラムから開く** 」を選びます。そこで、解凍した VCSSL ランタイムのフォルダから、以下のファイルを選択します。

- VCSSL_2.*.*.bat (コマンドラインバッチファイル)

このファイルを選択したら、「 **この種類のファイルを開くときは、選択したプログラムをいつも使う** 」の項目を有効にして、「 **OK** 」を押して完了します。すると、「 Test.vcssl 」が実行されます。

これで、インストール作業は完了です。以上のインストール作業を一度行った後は、拡張子が「 .vcssl 」のファイルをダブルクリックするだけで、すぐに実行できるようになります。

また、コマンドプロンプトなどでは、VCSSL プログラムのファイル名を入力するだけで、自動的に実行できるようになります。

※ Windows は米国 Microsoft Corporation の米国およびその他の国における登録商標です。

■ インストール作業 — Windows 以外の場合

Windows 以外でも、シェルスクリプトが動作する一般的なオペレーティングシステムでは、インストールして使用する事が可能です。

なお、ここでは **bash** シェルの利用を想定しています。それ以外のシェルをご利用の場合は、シェルスクリプトを適切な内容へ改変してご利用ください。

・フォルダの移動

まず、解凍した VCSSL ランタイムのフォルダを、適当な場所へ移動してください。ここでは「username」というユーザー名でログインしていて、以下のフォルダに移動したものとします。

```
/usr/local/bin/vcssl/vcssl_2_*_*
```

※2_*_*の部分は、利用するバージョンの番号に書き換えてください。

・実行権限の付加

ターミナルなどのコマンド入力端末を起動し、上で作成したフォルダまで cd コマンドで移動した状態で、以下のように入力してください。

```
chmod +x vcssl
```

・メモリー使用量の選択(任意)

標準状態では、VCSSL ランタイム実行時に割り当てられるメモリー使用量は少なめに限定されています。これを変更したい場合、以下の手順を実行してください。

まず、移動したフォルダー内にある、「**vcssl (拡張子無し)**」というファイルをテキストエディタで開いてください。内容は bash シェルスクリプトになっています。

内容の記述の後半に、下記のように「java -Xmx~m …」といった行が複数連続しています。これらの行の中で、VCSSL ランタイム実行時に割り当てたいメモリー使用量に該当する行のみ、先頭の「#」を削除してください。そして、最初から「#」の無かった「java -jar VCSSL.jar …」の行には「#」をつけておいてください。

```
#USE DEFAULT MEMORY
```

```
java -classpath ./bin/ -jar ./bin/VCSSL.jar $ABSOLUTE_FILE_PATH
```

```
#USE 64MB
```

```
#java -Xmx64m -classpath ./bin/ -jar ./bin/VCSSL.jar $ABSOLUTE_FILE_PATH
```

```
#USE 128MB
```

```
#java -Xmx128m -classpath ./bin/ -jar ./bin/VCSSL.jar $ABSOLUTE_FILE_PATH
```

```
#USE 256MB
```

```
#java -Xmx256m -classpath ./bin/ -jar ./bin/VCSSL.jar $ABSOLUTE_FILE_PATH
```

```
...
```

・環境変数 PATH への登録

ホームディレクトリにある「`.bashrc`（隠しファイル）」または「`.bash_profile`」もしくは「`.profile`」というファイル(※)をテキストエディタで開き、末尾に以下の一行を追記してください。この際、他の行を誤って改変しないようにご注意ください。

追記したら、上書き保存してテキストエディタを閉じ、ログアウトして再度ログインしてください。

```
export PATH=$PATH:/usr/local/bin/vcssl/vcssl_2_*/
```

※2_*の部分、利用するバージョンの番号に書き換えてください。

※これらのファイルはログイン時やシェル起動時に実行される設定ファイルですが、どのファイルが使用されるかはオペレーティングシステムの種類やバージョンによって異なります。

・プログラムの実行方法

以上で、インストール作業は完了です。

プログラムの実行は、コマンド入力端末から行います。実行コマンドは「`vcssl`」です。

例として、「`Test.vcssl`」を実行するには、以下のようにコマンド入力を行います。

```
vcssl Test.vcssl
```

■ プログラムの配布

VCSSL プログラムを実行するには、必ず VCSSL ランタイムが必要となります。そのため、RINEARN の VCSSL ランタイムでは、自分で開発した VCSSL プログラムに、VCSSL ランタイムを同梱して配布（頒布）する事を認めています。

ここでは、VCSSL ランタイムを同梱して、VCSSL プログラムを配布する際の、具体的な手順を解説しましょう。

・VCSSL ランタイムの同梱(任意)

VCSSL ランタイムを再配布（再頒布）するには、以下のような条件が付きます。

- ・再配布可能なのは VCSSL.jar のみ。
(RinearnGraph3D.jar や RinearnGraph2D.jar の再配布は NG)
- ・同梱する VCSSL プログラムを実行するという目的でのみ、再配布が可能。
(VCSSL ランタイム単体での再配布は NG)
- ・「RINEARN VCSSL Runtime 使用許諾契約書(RINEARN_VCSSL_Runtime_License.txt)」も同梱するか、その内容を同梱の文書に掲載する必要がある。
- ・同梱する VCSSL ランタイムに関する一切の責任を、VCSSL ランタイムの開発元に負担させない事に同意する必要がある。

他にもいくつかの細かい条件がありますが、詳しくは「RINEARN VCSSL Runtime 使用許諾契約書(RINEARN_VCSSL_Runtime_License.txt)」をご参照ください。

・起動ファイルの作成

VCSSL ランタイムを同梱するだけでは、ユーザーが VCSSL プログラムを起動する際に、まず VCSSL ランタイムを起動して、それから VCSSL プログラムを選択しなければいけません。これでは少し不便です。そこで、VCSSL ランタイムを同梱する場合は、一緒に起動ファイルを作成して同梱するのがおすすめです。

起動ファイルにはいくつかの種類があります。最も簡単な起動ファイルを作成するだけでも、自動的に特定の VCSSL プログラムが起動されるようになり、これだけでかなり便利になります。

さらに高度な機能（ 任意の拡張子のファイルをダブルクリックすると、自動的に VCSSL プログラムが起動し、そのプログラムでファイルを開くようにするなど ）を実現するには、オペレーティングシステムの種類に応じた、高度な起動ファイルを作成します。これについても後述します。

・簡単な起動ファイルの作成

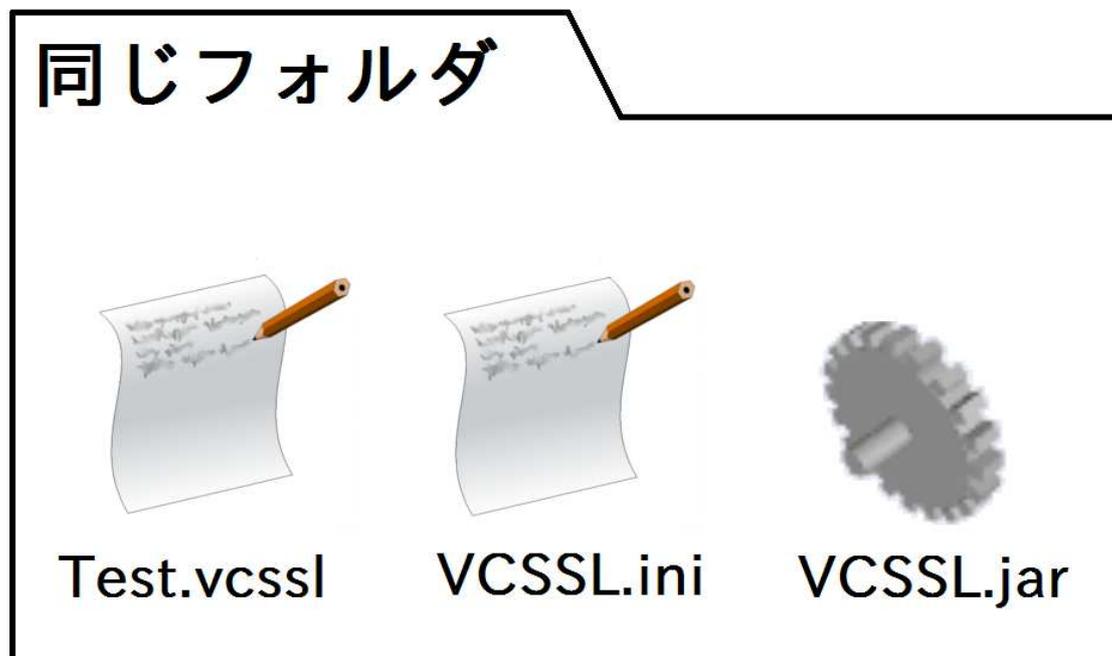
まずは、特定の VCSSL プログラムを自動で起動するだけの、最も簡単な起動ファイルを作成してみましょう。

メモ帳などのテキストエディタを開き、下記のように記述してください。

```
PROGRAM = Test.vcssl
```

※Test.vcssl の箇所は、自作の VCSSL プログラム名に変えてください。

記述できたら、「 VCSSL.ini 」という名前（ ファイル名は任意ですが、拡張子は必ず.ini にする必要あり ）で、ファイルの種類に「 **すべてのファイル** 」を選択して保存してください。保存したファイルの説明欄に「 **構成設定ファイル** 」などと表示されていれば、正常に作成できています。



ここで作成した「 Test.bat 」と、「 VCSSL.ini 」、および「 VCSSL.jar 」を同じフォルダに入れて、VCSSL.jar をダブルクリックすると、Test.vcssl が自動で実行されます。このように、起動時にユーザーがいちいちプログラムを選択する必要がなくなります。

・高度な起動ファイルの作成 (Windows の場合)

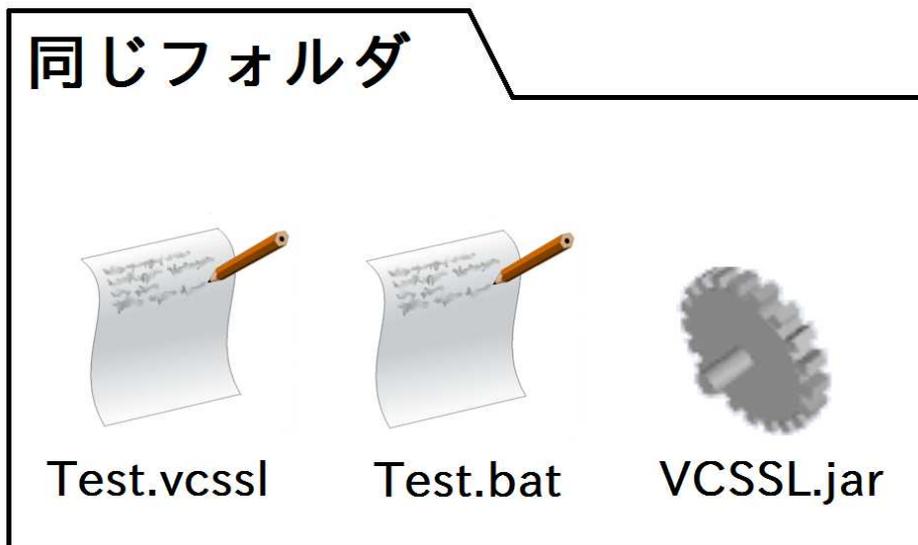
続いて、Windows を対象とした、高度な起動ファイルについて解説しましょう。ここでは「 Test.vcssl 」を起動する起動ファイルを作成します。

メモ帳などのテキストエディタを開き、下記のように記述してください。

```
cd /d %~dp0
java -jar VCSSL.jar Test.vcssl %1
```

※最後の行の Test.vcssl は、自作の VCSSL プログラム名に変えてください。

記述できたら、「 Test.bat 」という名前 (ファイル名は任意ですが、拡張子は必ず.bat にする必要あり) で、ファイルの種類に「 **すべてのファイル** 」を選択して保存してください。保存したファイルの説明欄に「 **バッチファイル** 」などと表示されていれば、正常に作成できています。



ここで作成した「 Test.bat 」と、「 Test.vcssl 」、および「 VCSSL.jar 」を同じフォルダに入れて、Test.bat をダブルクリックすると、Test.vcssl が自動で実行されます。

また、任意のファイルを右クリックして、「プログラムから開く」から「 Test.bat 」を選択すると、「 Test.vcssl 」の main 関数の引数に、そのファイルのパスが送られます (main 関数がない場合は無視される)。

このように、VCSSL プログラムを、一般のソフトウェアと全く同様に使うことが可能となり、大変便利になります。

※ Windows 以外のオペレーティングシステムでこのファイルを作成する場合は、文字コードに「 Shift_JIS」、改行コードに「 CR+LF 」を選択する必要があります。

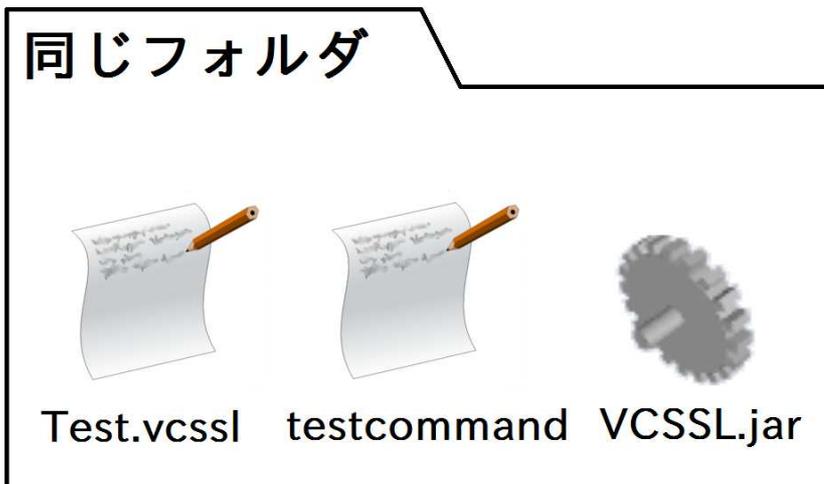
・高度な起動ファイルの作成（ Windows 以外の場合 ）

最後に、Windows 以外を対象とした、高度な起動ファイルについて解説しましょう。引き続き、「 Test.vcssl 」を起動する起動ファイルを作成します。また、ここでは **bash シェル**を使用した環境を想定します。それ以外のシェルを使用している場合は、記述内容を適切に変更してください。

まず、適当なテキストエディタを開き、下記のように記述し、「 **testcommand** 」という名前（ 任意 ）で保存してください。

```
CURRENT_DIRECTORY_PATH=$(pwd)
ABSOLUTE_FILE_PATH=$(pwd)/$1
SHELL_SCRIPT_PATH=$(dirname $0)
cd $SHELL_SCRIPT_PATH
java -jar VCSSL.jar Test.vcssl $ABSOLUTE_FILE_PATH
```

※最後の行の Test.vcssl は、自作の VCSSL プログラム名に変えてください。



ここで作成した「 testcommand 」と、「 Test.vcssl 」、および「 VCSSL.jar 」を同じフォルダに入れて、bash の環境変数「PATH」に、そのフォルダのパスを登録しておけば、コマンド入出力端末のどこからでも、「 testcommand 」とコマンド入力するだけで「 Test.vcssl 」が実行されるようになります。

また、コマンド引数は、「 Test.vcssl 」の main 関数の引数に、そのファイルのパスが送られます（ main 関数がない場合は無視される ）。

なお、bash へのパス登録作業や、実行権限の付加作業などについては、本書 12 ページ「インストール作業 - Windows 以外の場合 」を参考に、取り扱い説明書などでご説明ください。

※ Windows でこのファイルを作成する場合は、文字コードに「 UTF-8 」、改行コードに「 LF 」を選択する必要があります。

・必要な文書などを同梱し、圧縮する

最後に、上で用意したフォルダに、「RINEARN VCSSL Runtime 使用許諾契約書 (RINEARN_VCSSL_Runtime_License.txt)」と、プログラムの取り扱い説明書 (Readme.txt など) を作成して同梱しましょう。ただし、前者は VCSSL ランタイムを同梱しない場合には不要です。

なお、プログラムの取り扱い説明書では、VCSSL ランタイムを使用する前に「RINEARN VCSSL Runtime 使用許諾契約書」を確認するよう、ユーザーに対してご説明ください。

上記の文書や、その他のファイルなどを全て同梱したら、フォルダごと ZIP ファイルなどに圧縮しましょう。圧縮した ZIP ファイルの名前には、プログラム名、およびバージョンなどを分かりやすく記載しておく事をおすすめします。

test_1_0_0.zip

これで、配布の準備は全て整いました。あとは、ブログやホームページ、ソフトウェアのダウンロードサイト(ストア)などで自由に配布しましょう。

・VCSSL ランタイムの同梱と著作権に関して

自分で開発した VCSSL プログラムに関する著作権は、その開発者に帰属します。その VCSSL プログラムについて、RINEARN は何の権利も持ちません。つまり、その VCSSL プログラムを単体で配布する場合には、配布物は 100% 開発者の著作物であり、完全に自由に扱えます。

しかしながら、その VCSSL プログラムに、VCSSL ランタイムを同梱して配布する際には注意が必要です。RINEARN の VCSSL ランタイムに関する著作権は、RINEARN に帰属します。つまり、VCSSL ランタイムを同梱して配布する場合は、配布物に RINEARN の著作物が含まれる事になります。

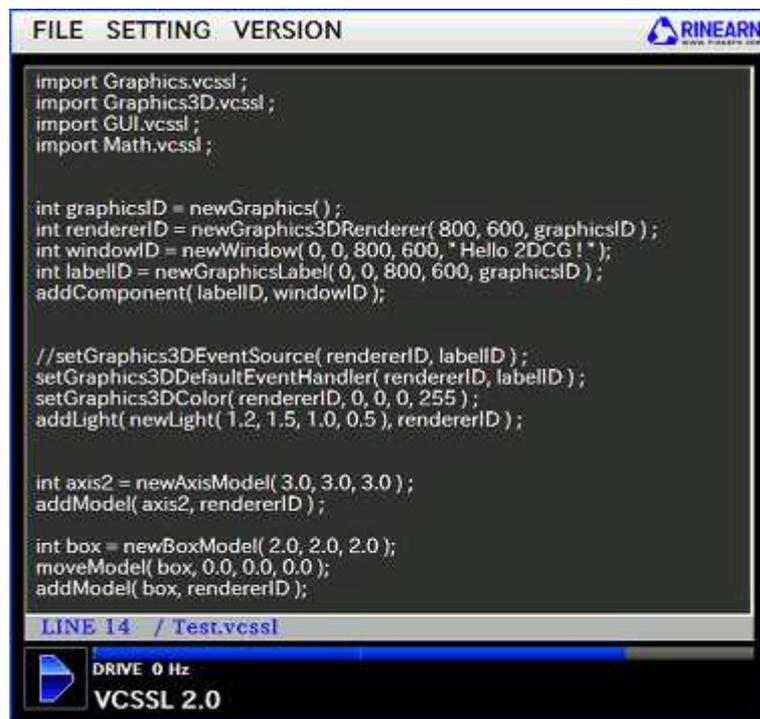
何らかの事情により、この事を避けたい場合があるかもしれません。そういった場合は、VCSSL ランタイムを同梱せずに配布し、ユーザーに自分で VCSSL ランタイムを入手するようにご説明ください。

■ 統合開発環境

これまでの流れでは、テキスト記述用のソフトでプログラムを記述し、それを VCSSL ランタイムで実行するという方法をとってきました。しかし、それを一つのソフトウェア上で一括して行えると便利です。

こういった、プログラミングに必要な一連の作業をまとめて行えるようなソフトウェアは、一般に統合開発環境と呼ばれます。VCSSL にも、統合開発環境として「リニアンコーダー (RINEARN Coder)」が用意されています。

リニアンコーダーでは、テキストエディタ部分にプログラムを書き、画面左下のスタートボタンを押すだけで、すぐに実行できます。



The screenshot shows the RINEARN Coder IDE interface. The title bar reads "FILE SETTING VERSION" and the RINEARN logo is in the top right. The main editor area contains the following code:

```
import Graphics.vcssl ;
import Graphics3D.vcssl ;
import GUI.vcssl ;
import Math.vcssl ;

int graphicsID = newGraphics( ) ;
int rendererID = newGraphics3DRenderer( 800, 600, graphicsID ) ;
int windowID = newWindow( 0, 0, 800, 600, "Hello 2DCG!" ) ;
int labelID = newGraphicsLabel( 0, 0, 800, 600, graphicsID ) ;
addComponent( labelID, windowID ) ;

//setGraphics3DEventSource( rendererID, labelID ) ;
setGraphics3DDefaultEventHandler( rendererID, labelID ) ;
setGraphics3DColor( rendererID, 0, 0, 0, 255 ) ;
addLight( newLight( 1.2, 1.5, 1.0, 0.5 ), rendererID ) ;

int axis2 = newAxisModel( 3.0, 3.0, 3.0 ) ;
addModel( axis2, rendererID ) ;

int box = newBoxModel( 2.0, 2.0, 2.0 ) ;
moveModel( box, 0.0, 0.0, 0.0 ) ;
addModel( box, rendererID ) ;
```

Below the code, it says "LINE 14 / Test.vcssl". At the bottom left, there is a blue play button icon and the text "DRIVE 0 Hz" and "VCSSL 2.0".

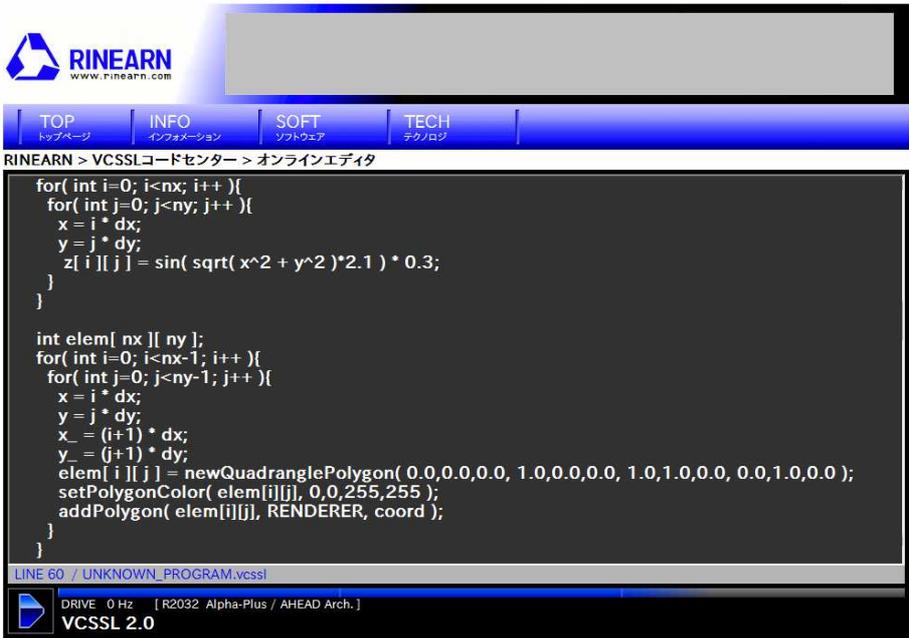
<http://www.rinearn.com/coder/>

▲ リニアンコーダー 公式 WEB サイト URL

■ WEB で開発 / 実行も可能

VCSSLでは、WEB ブラウザ上でプログラムの開発と実行を行う事ができる、「VCSSLオンラインプレイヤー」というサービスも用意されています。

登録やログインなどの面倒な作業は一切不要で、下記 WEB ページにアクセスするだけで、いつでも使用できます。



<http://www.rinearn.com/code/play/>

▲ VCSSL オンラインプレイヤー URL
上記 URL にアクセスするだけで、いますぐ使用できます

■ ブログや WEB ページでの公開に便利な VCSSL to HTML

VCSSL で開発したプログラムの中身を、ブログなどの WEB ページに掲載する際には、改行や特殊文字のエスケープ処理などを行わなければ、正常に表示できません。そこで、下記の WEB ツール「 VCSSL to HTML 」を使用すると便利です。VCSSL to HTML は、下記 URL にアクセスするだけで、WEB ブラウザ上でそのまま使用できます。

<http://www.rinearn.com/vcssltohtml/>

▲ VCSSL to HTML URL

VCSSL to HTML は、VCSSL プログラムを入力すると、それを WEB ページに掲載可能な HTML コードに自動変換して出力してくれるツールです。変換過程では、改行や特殊記号のエスケープ処理などの面倒な作業を自動で行ってくれるだけでなく、文法的に重要な語句・構文をカラーリングしたり、API 関数仕様へのハイパーリンクを付加したりする事ができます。

また、**ワンクリックでプログラムの内容をコピーし、VCSSL オンラインプレイヤーを起動して実行する事ができる**パネルも付けられます。これにより、WEB ページの訪問者は、掲載されているプログラムを WEB ブラウザ上でそのまま実行する事ができ、とても便利です。

■ VCSSL をはじめよう !

これで、VCSSL スタートアップガイドの内容はすべて終わりです。

このままプログラミングガイドへと進み、より実践的な VCSSL プログラミングをはじめましょう !



VCSSL公式ガイド
基礎文法
(PDF版)

▶ [いますぐダウンロード](#)

http://www.rinearn.com/vcssl/grammar/VCSSL_Syntax.pdf



VCSSL公式ガイド
GUI
(PDF版)

▶ [いますぐダウンロード](#)

http://www.rinearn.com/vcssl/gui/VCSSL_GUI.pdf



VCSSL公式ガイド
Graphics2D
(PDF版)

▶ [いますぐダウンロード](#)

http://www.rinearn.com/vcssl/graphics2d/VCSSL_Graphics2D.pdf



VCSSL公式ガイド
Graphics3D
(PDF版)

▶ [いますぐダウンロード](#)

http://www.rinearn.com/vcssl/graphics3d/VCSSL_Graphics3D.pdf

商標

[1] Oacle と Java は、Oracle Corporation 及びその子会社、関連会社の米国及びその他の国における登録商標です。文中の社名、商品名等は各社の商標または登録商標である場合があります。

[2] Windows は米国 Microsoft Corporation の米国およびその他の国における登録商標です。

VCSSL 2.0 スタートアップガイド 第9版

発行 2012年4月10日

著者 松井文宏

この文書の内容は、以下の Web サイトにおいても公開されています。

<http://www.rinearn.com/vcssl/>