

ソフトウェアインストール手順

1 目次

2	使用する PC の仮想化関連機能確認.....	4
3	VMware Workstation Player 15.5 のインストール.....	7
4	Ubuntu 18.04 のディスクイメージファイルのダウンロード	14
5	VMware への Ubuntu 18.04 インストール.....	15
6	Choreonoid のインストール	23

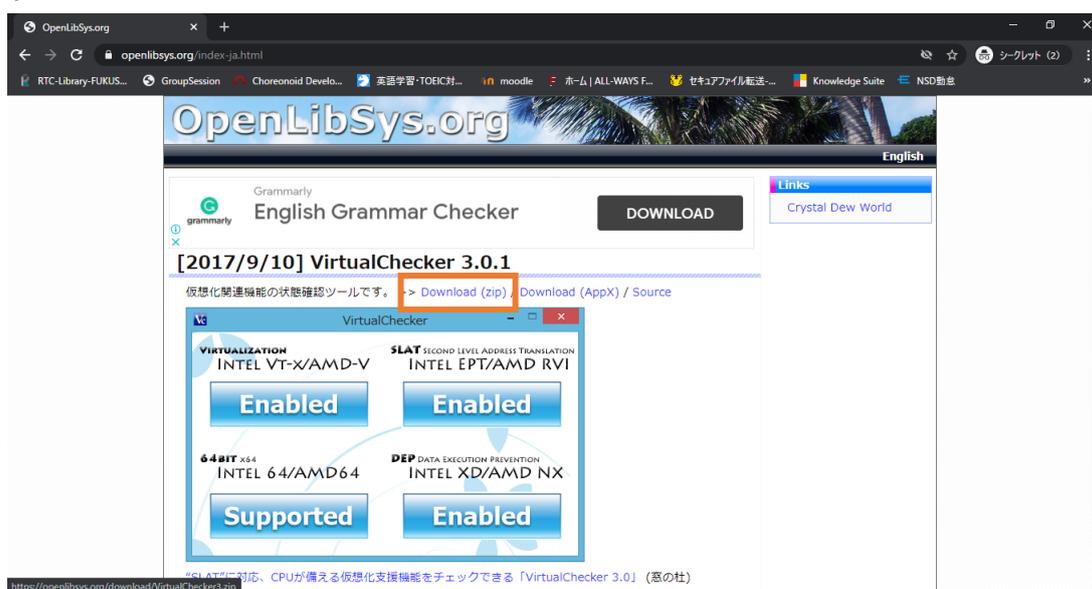
2 使用する PC の仮想化関連機能確認

まずは、現在使用している PC が VMware を使用できるかの確認を行います。

以下 URL より Virtualchecker をダウンロードし、仮想化関連機能の状況を確認します。

Virtualchecker : <https://openlibsys.org/index-ja.html>

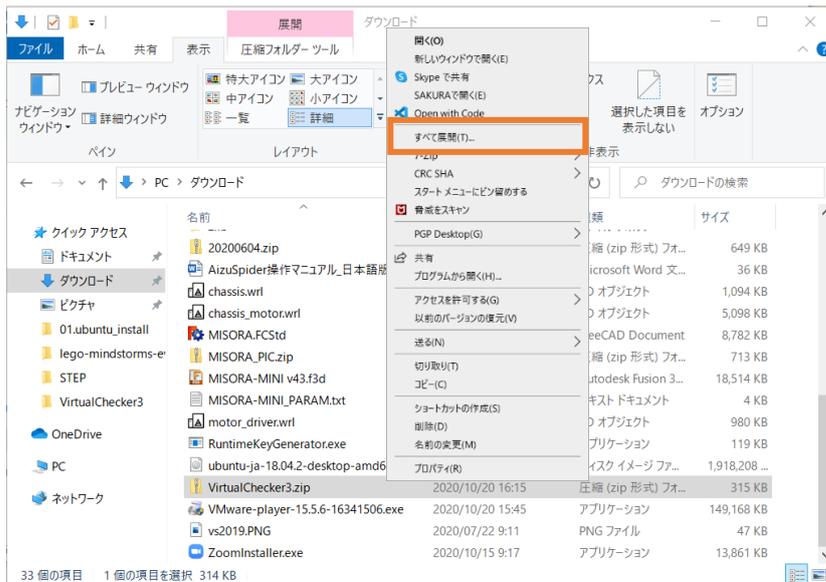
1. 「Download (zip)」リンクを押下すると、『VirtualChecker3.zip』がダウンロードされます。



Virtual Checker ダウンロードサイト

2. ダウンロードフォルダ^{※1}を開き、『VirtualChecker3.zip』上で右クリックし『すべて展開』を選択して解凍します。

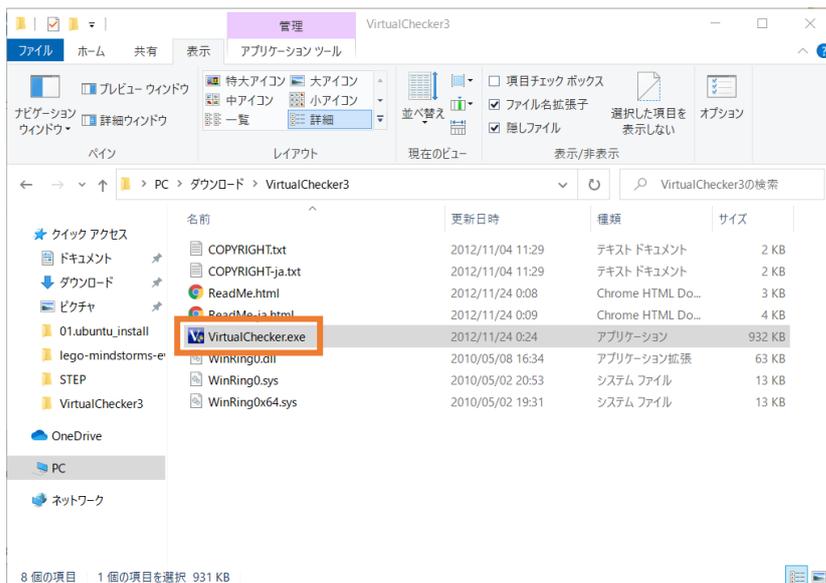
※1 ダウンロードフォルダは設定を変更していなければ、『C:¥Users¥ユーザー名¥Downloads』となります。



Virtual Checker の解凍

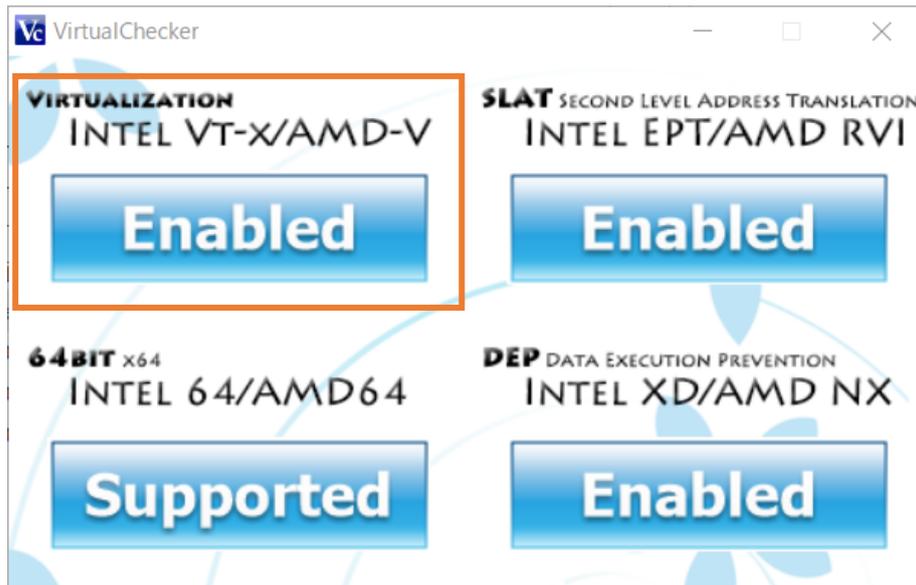
3. 解凍した『VirtualChecker3』フォルダ内の『VirtualChecker.exe』※2 を起動します。

※2 拡張子を表示する設定がオフになっている場合, “.exe”が表示されませんのでご注意ください。



Virtual Checker の実行

4. 起動すると以下のような画面が表示されるので, 『INTEL VT-X/AMD-V』が”Enabled”であることを確認してください。



Virtual Checker による仮想化環境の確認

5. “Disabled”の場合は、BIOS の設定から『INTEL VT-X/AMD-V』を有効化してください。
“Unsupported”^{※3}の場合、未対応の PC となるため対応する CPU の PC を使用してください。

※3 Windows10 に Choreonoid をインストールして動かすことも可能なので、未対応の方はご連絡ください。

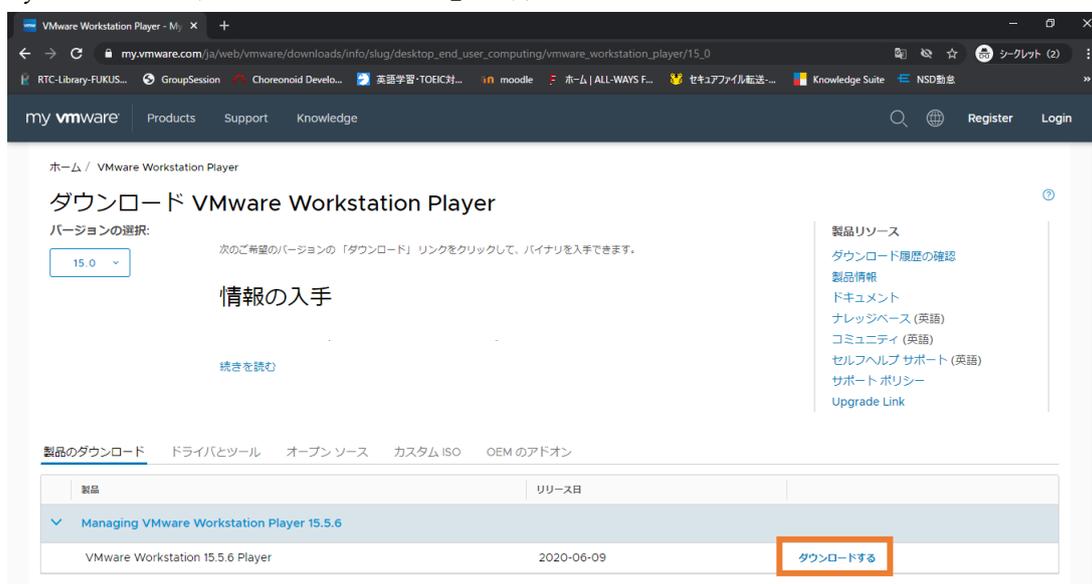
3 VMWARE WORKSTATION PLAYER 15.5 のインストール

以下 URL より VMware Workstation Player をダウンロードし、インストールを行います。
本講習では、バージョン 15.5 のものを使用します。

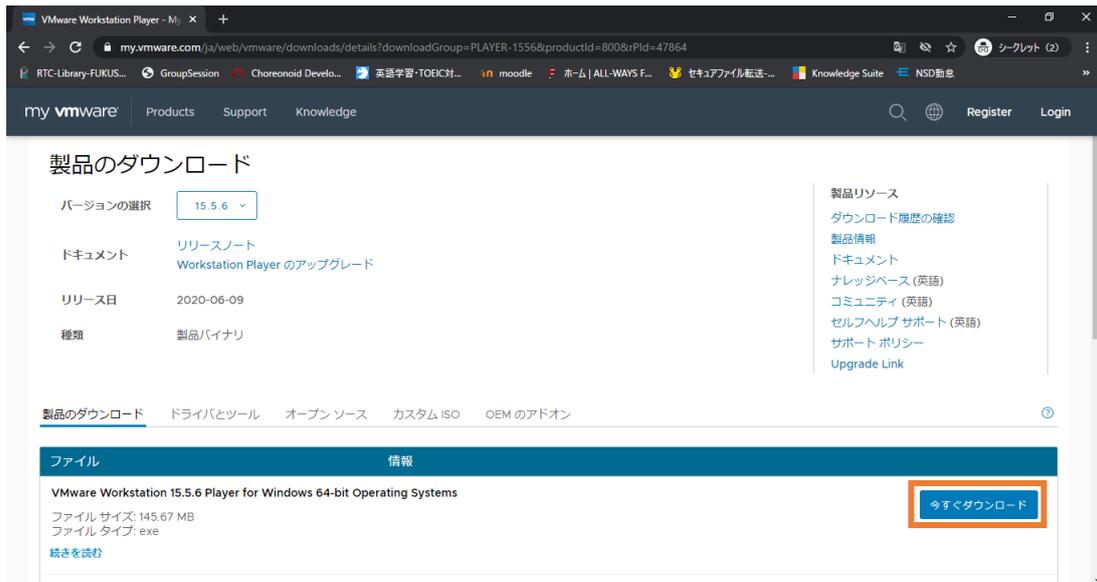
VMware Workstation Player :

https://my.vmware.com/ja/web/vmware/downloads/info/slug/desktop_end_user_computing/vmware_workstation_player/15_0

1. 『ダウンロードする』リンクをクリックすると、バイナリファイルのダウンロードページに遷移するので、“VMware Workstation 15.5.6 Player for Windows 64-bit Operating Systems”の『今すぐダウンロード』を押下します。

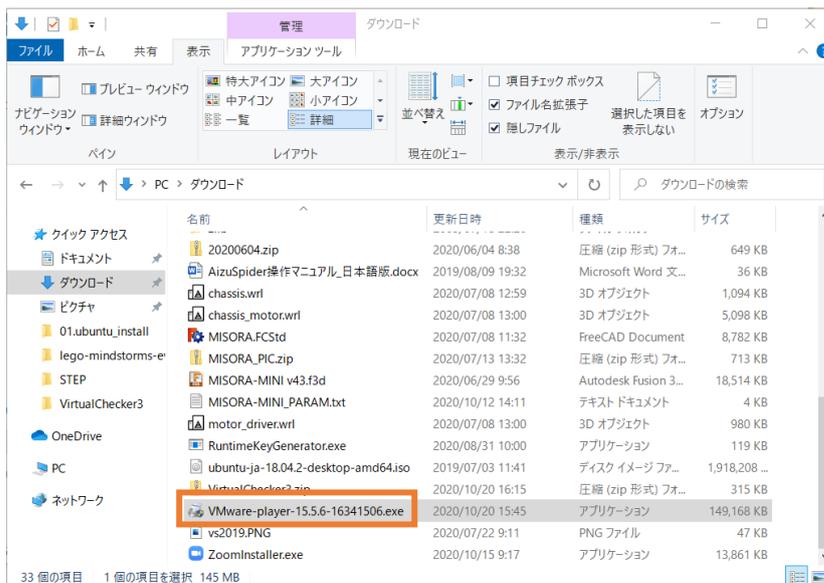


VMware Workstation Player のダウンロードサイト



VMware Workstation Player のダウンロード

2. ダウンロードフォルダを開き、『VMware-player-15.5.6-16341506.exe』をダブルクリックし実行します。



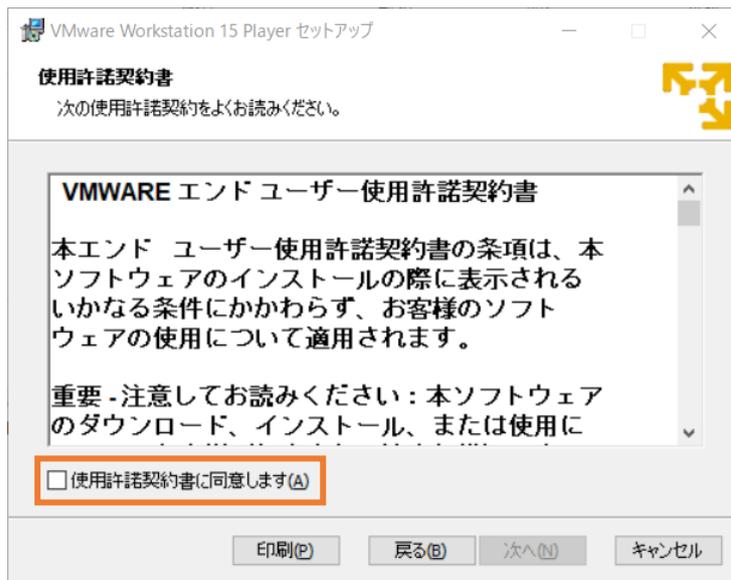
VMware Workstation Player のインストーラを実行

3. 『このアプリがデバイスに変更を加えることを許可しますか?』ポップアップが表示されるので、『はい』を押下します。
4. 『VMware Workstation 15 Player セットアップウィザード』が表示されるので、『次へ』ボタンを押下します。

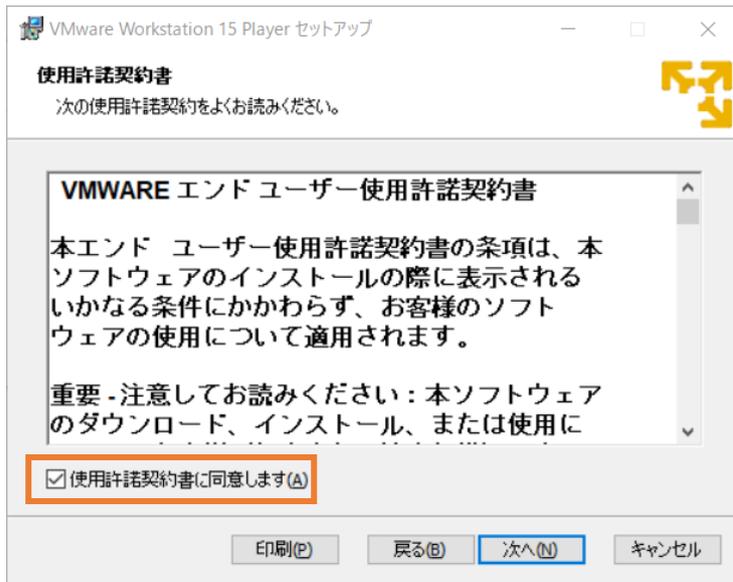


VMware Workstation Player セットアップウィザード

- 『使用許諾契約書』が表示されるので、『使用許諾契約書に同意します』にチェックをいれ、『次へ』ボタンを押下します。

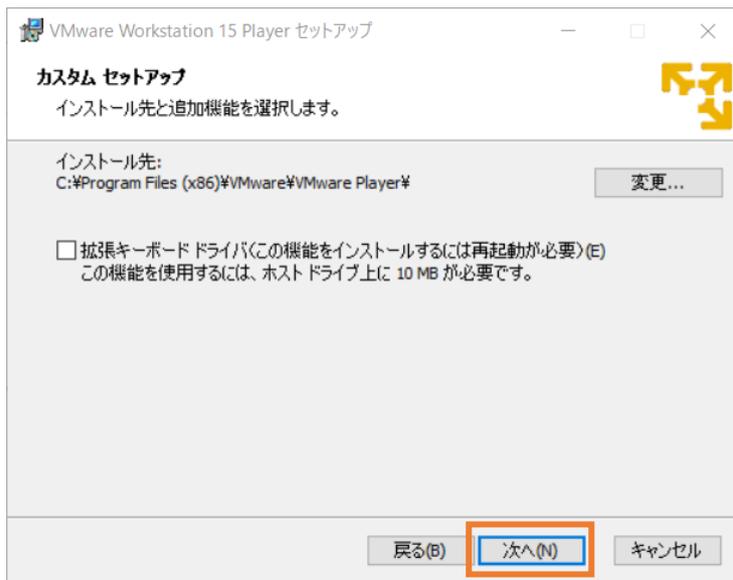


使用許諾契約書（チェック前）



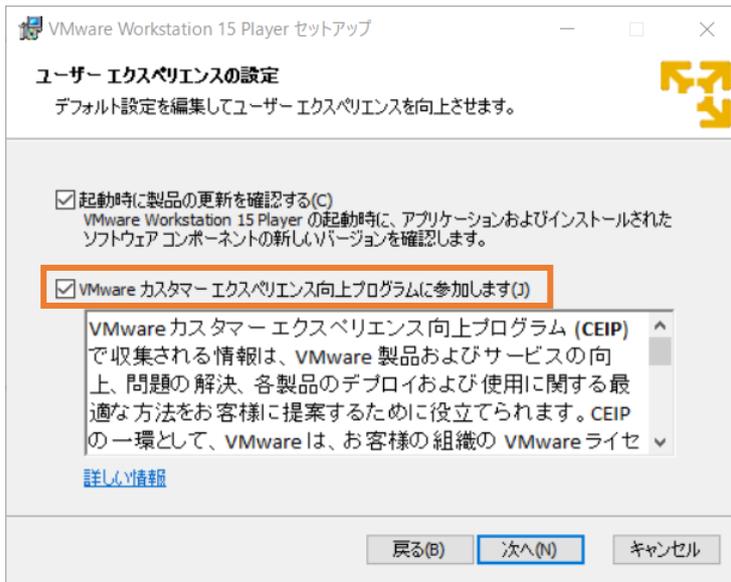
使用許諾契約書（チェック後）

- 『カスタムセットアップ』が表示されるので、初期設定の状態で『次へ』ボタンを押下します。

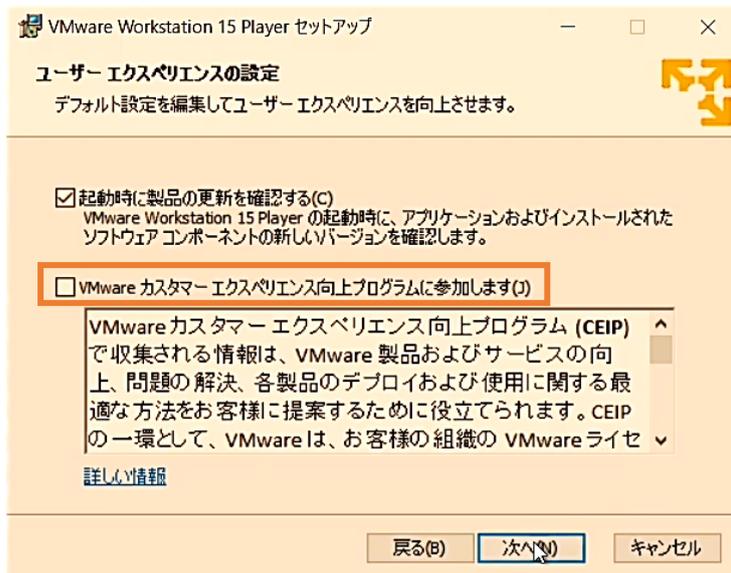


カスタムセットアップ

- 『ユーザーエクスペリエンスの設定』が表示されるので、『VMware カスタマーエクスペリエンス向上プログラムに参加します』のチェックを外し『次へ』ボタンを押下します。

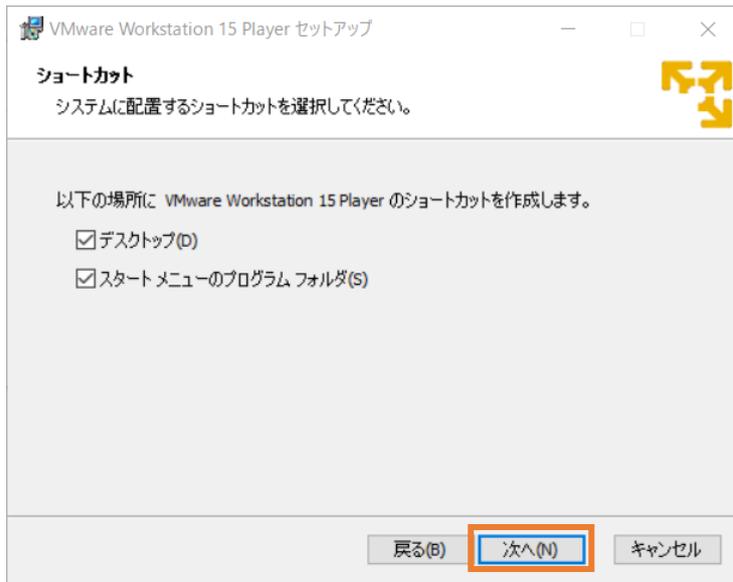


ユーザーエクスペリエンスの設定（チェック解除前）



ユーザーエクスペリエンスの設定（チェック解除後）

- 『ショートカット』が表示されるので、初期設定の状態ですら『次へ』ボタンを押下します。



ショートカット

- 『VMware Workstation 15 Player のインストール準備完了』が表示されるので、『インストール』ボタンを押下します。



インストール準備完了

- 『VMware Workstation 15 Player セットアップウィザードが完了しました』が表示されるので、『完了』ボタンを押下します。



セットアップ完了

4 UBUNTU 18.04 のディスクイメージファイルのダウンロード

以下 URL より Ubuntu 18.04 のディスクイメージファイルをダウンロードします。

Ubuntu 18.04 : <https://www.ubuntulinux.jp/products/JA-Localized/download>

『ubuntu-ja-18.04.3-desktop-amd64.iso (ISO イメージ)』リンクをクリックし、ダウンロードします。



Ubuntu 18.04 ディスクイメージダウンロードサイト

5 VMWARE への UBUNTU 18.04 インストール

スタートメニューもしくはデスクトップ上に表示されている『VMware Workstation 15 Player』から VMware を起動し、Ubuntu 18.04 のインストールを行います。

1. 『非営利目的で VMware Workstation 15 Player を無償で使用する』を選択し、『続行』ボタンを押下します。



ライセンスの設定

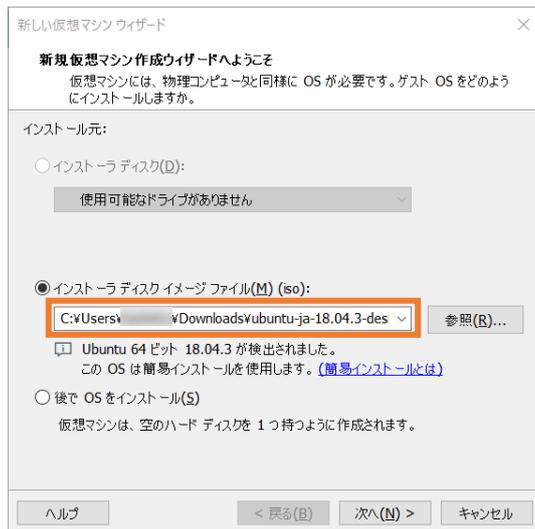


ライセンス設定完了

2. 『ソフトウェアの更新』ダイアログが表示されるので、『このバージョンをスキップ』ボタンを押下します。



インストーラディスクイメージファイルの選択



Ubuntu 18.04 のディスクイメージファイルの選択

5. “フルネーム”, “ユーザ名”, “パスワード”, “確認”^{※2}を入力し, 『次へ』 ボタンを押下します.
※2 各自で決めた適当なものを入力します.

新しい仮想マシンウィザード

簡易インストール情報
これは Ubuntu 64 ビット のインストールに使用します。

Linux のパーソナライズ

フルネーム(F):

ユーザー名(U):

パスワード(P):

確認(C):

ヘルプ < 戻る(B) 次へ(N) > キャンセル

インストール情報の入力

- 『仮想マシン名』を“Ubuntu 18.04”に変更し、『場所』はデフォルトの状態です。『次へ』ボタンを押下します。

新しい仮想マシンウィザード

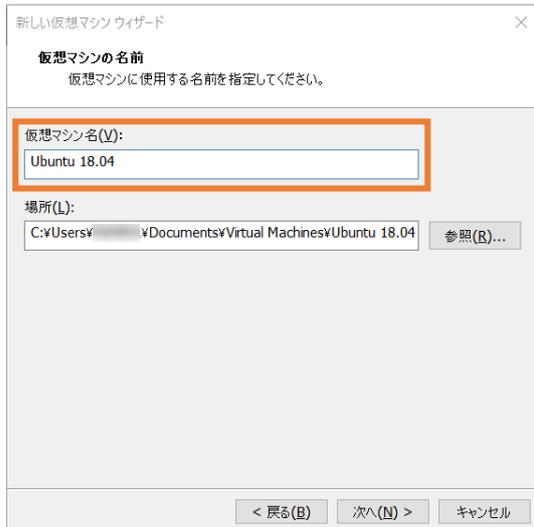
仮想マシンの名前
仮想マシンに使用する名前を指定してください。

仮想マシン名(N):

場所(L):
 参照(R)...

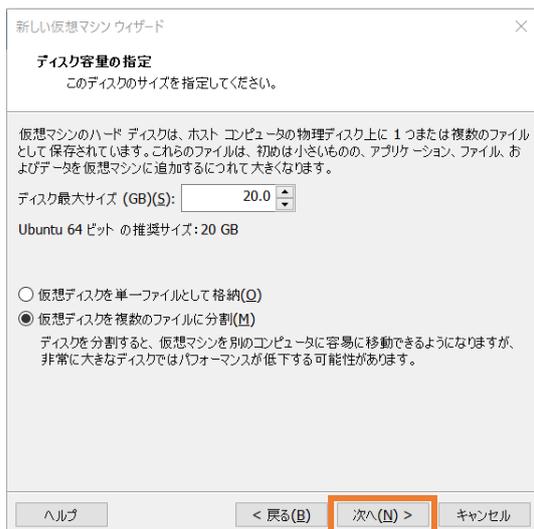
< 戻る(B) 次へ(N) > キャンセル

仮想マシン名の設定（変更前）



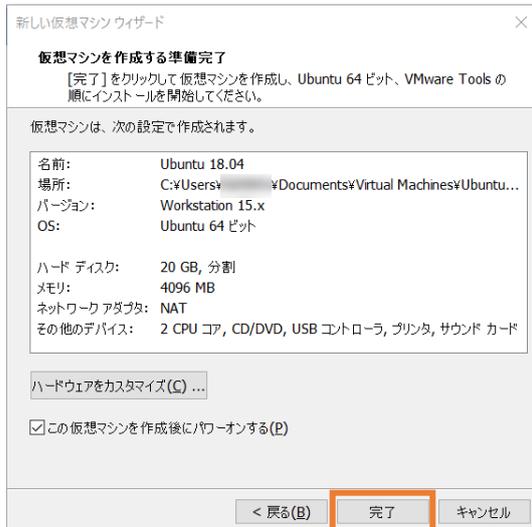
仮想マシン名の設定 (変更後)

- 『ディスク最大サイズ』は、“20GB”を指定し、『仮想ディスクを複数のファイルに分割』を選択し『次へ』ボタンを押下します。



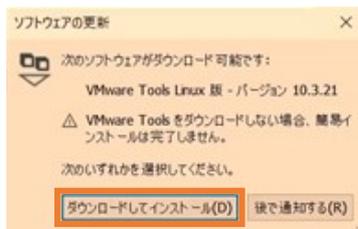
ディスク容量の指定

- 『完了』ボタンを押下し, Ubuntu 18.04 のインストールを開始します。



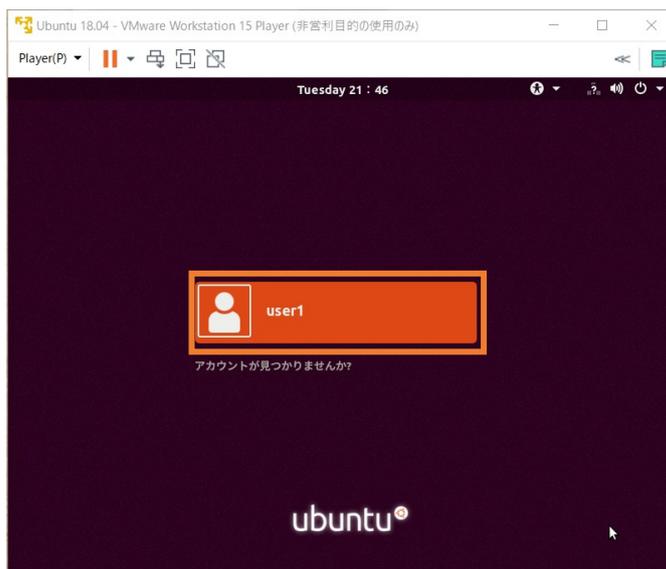
仮想マシンを作成する準備完了

- 『ソフトウェアの更新』ダイアログが表示され、VMware Tools Linux 版をインストールするか確認されるので、『ダウンロードしてインストール』ボタンを押下します。

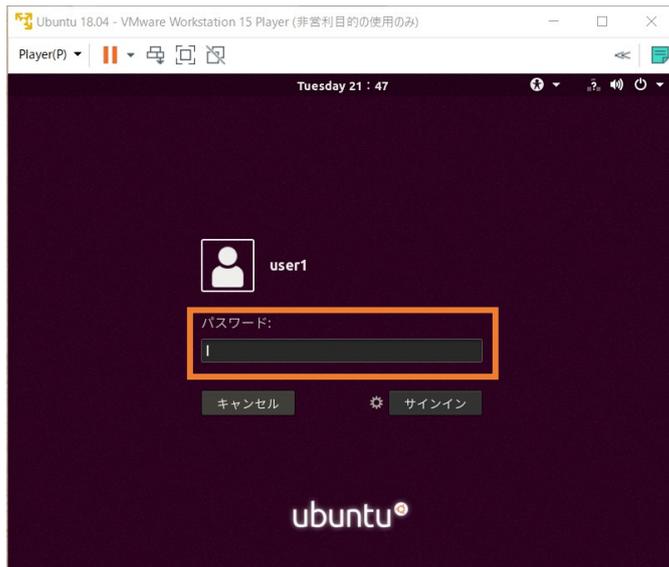


ソフトウェアの更新

- インストールが完了したら、ユーザ名をクリックし項番 3 で設定したパスワードを入力します。

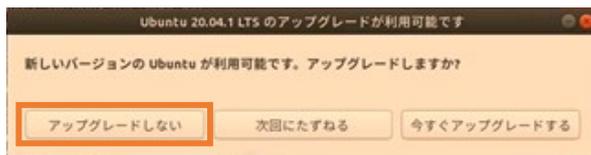


Ubuntu ログイン画面



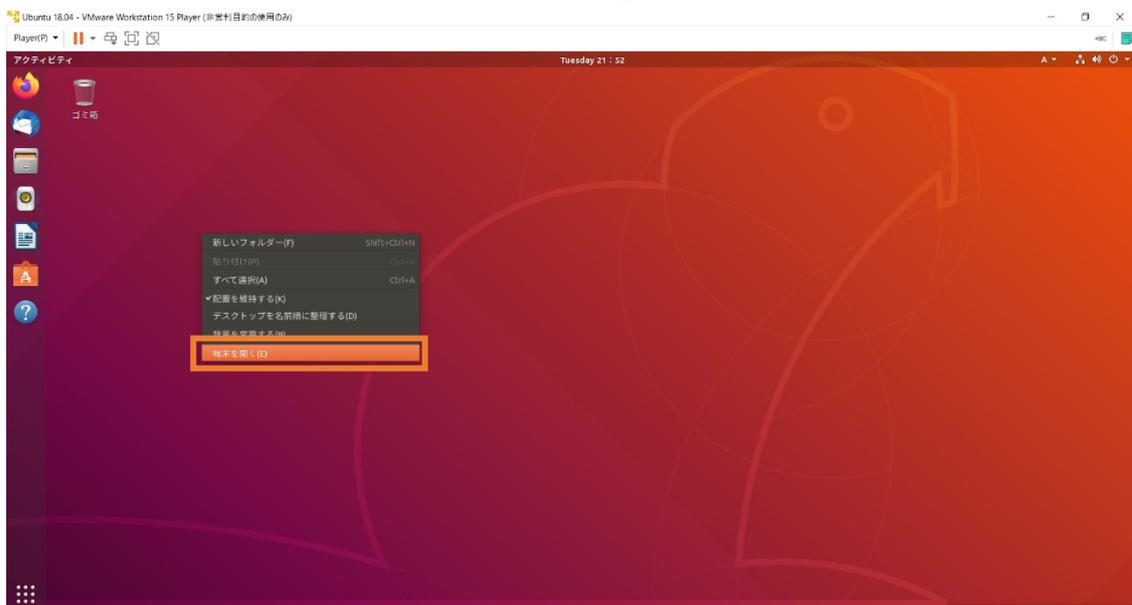
パスワード入力画面

11. 『Ubuntu 20.04.1 LTS のアップグレードが利用可能です』 ポップアップが表示されるので、『アップグレードしない』 ボタンを押下します。

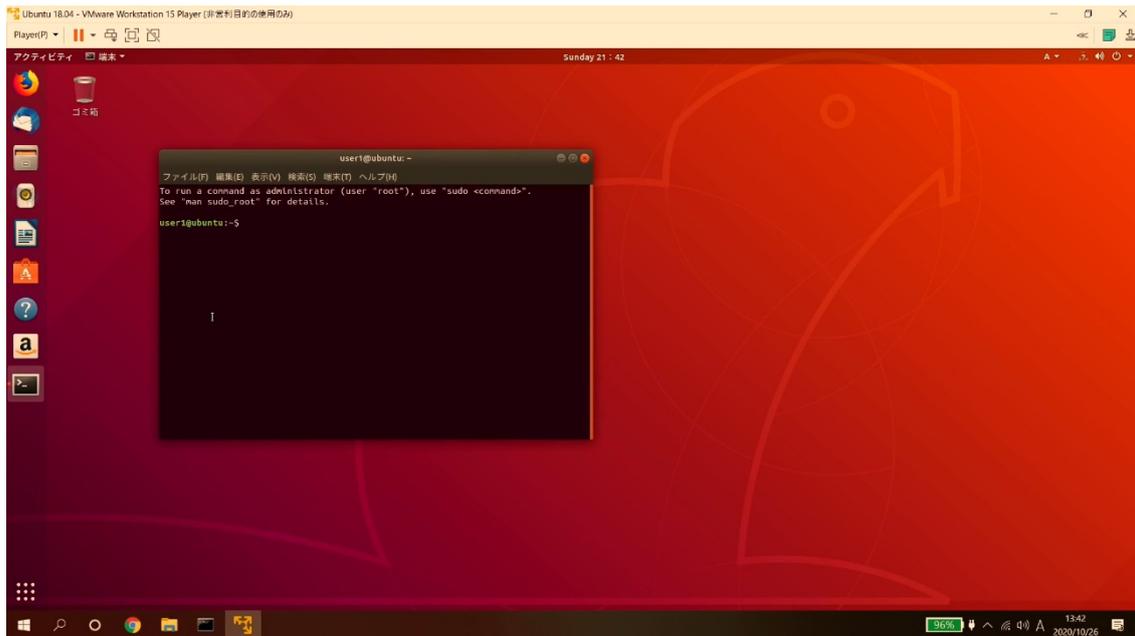


Ubuntu アップグレード確認

12. デスクトップ上で右クリックし、『端末を開く』を押下します。



端末を開く



13. ターミナル上で以下のコマンドを実行し、パッケージ一覧の更新を行います。
`$ sudo apt update`
14. 以下のコマンドを実行し、パッケージの更新を行います。
`$ sudo apt upgrade -y`
15. 再起動を行うか確認されるので、『すぐに再起動』ボタンを押下し再起動します。

6 CHOREONOID のインストール

今回は Git より開発版をダウンロードしインストールを行います。

1. デスクトップ上で右クリックし、『端末を開く』を押下します。

2. Choreonoid リポジトリをダウンロードするために、git のインストールを行います。

```
$ sudo apt install -y git
```

3. Choreonoid リポジトリをダウンロードします。TurtleBot 関連のプログラムの動作を確認できているのは、2020/12/18 時点までです。それ以降は、動作未検証のため、正常に動作しないことがあります。そのため、クローンを行った後に、チェックアウトのコマンドを実行し、2020/12/18 時点のソースを取得してください。

```
$ cd ~/
$ git clone https://github.com/choreonoid/choreonoid.git
$ cd ~/choreonoid
$ git checkout 46f1284a1b8cab9571e2b6d3d1e220912a1846a6
```

4. 依存パッケージのインストールを行います。“~/choreonoid/misc/script”ディレクトリ内に“install-requisites-ubuntu-18.04.sh”というスクリプトがあるので、それを実行します。

```
$ cd ~/choreonoid
$ misc/script/install-requisites-ubuntu-18.04.sh
```

5. TurtleBot 関連のファイルのダウンロードを行います。以下の GitHub リポジトリの取得を行います。『リポジトリ全体の取得』もしくは『一部のディレクトリの取得』のいずれかの方法でリポジトリの取得を行ってください。

GitHub URL : <https://github.com/RTC-Library-FUKUSHIMA/Education.git>

```
$ cd ~/choreonoid/ext
```

【リポジトリ全体の取得】

リポジトリ全体をクローンします。

```
$ git clone https://github.com/RTC-Library-FUKUSHIMA/Education.git
```

上記 URL の 3DCAD モデルは、以下サイトのファイルのままですと Choreonoid で表示したときに、テクスチャが崩れる現象を確認したため、正常に表示されるように VRML 形式に変換したものです。

turtlebot のプレート類 : <https://github.com/turtlebot/turtlebot.git>

kobuki : <https://github.com/yujinrobot/kobuki.git>

autorace : https://github.com/ROBOTIS-GIT/autorace_track.git

【一部のディレクトリの取得】

Education ディレクトリを生成します。

```
$ mkdir Education
```

Education ディレクトリに移動します。

```
$ cd Education
```

リポジトリを初期化します。

```
$ git init
```

一部のディレクトリ取得のために sparseCheckout を設定します。

```
$ git config core.sparseCheckout true
```

Git の設定ファイルの確認を行います。

```
$ cat .git/config
```

取得元のリポジトリの設定をします。

```
$ git remote add origin https://github.com/RTC-Library-FUKUSHIMA/Education.git
```

取得したいディレクトリを sparseCheckout に設定します。

```
$ echo ChoreonoidWorkshop > .git/info/sparse-checkout
```

リモートリポジトリからデータを取得します。

```
$ git pull origin master
```

ChoreonoidWorkshop ディレクトリが取得できたことを確認します。

```
$ ls
```

6. リポジトリが取得できたら、CMakeLists.txt の生成を行います。

ChoreonoidWorkshop ディレクトリに移動します。

```
$ cd ChoreonoidWorkshop
```

ChoreonoidWorkshop ディレクトリをビルド対象に含めるためにシェルスクリプトを実行し、MakeLists.txt を生成します。

```
$ ./createCMakeLists.sh
```

上の階層へ移動し、CMakeLists.txt が生成されていることを確認します。

```
$ cd ../
```

\$ 1s

7. 必要なファイル類のダウンロード, インストールが完了したので, ビルドを行います.
ビルドに使用するディレクトリ, Choreonoid をインストールするディレクトリを作成します.

```
$ cd ~/choreonoid
$ mkdir build
$ mkdir program
```

8. ビルドディレクトリ上で CMake を実行します. Choreonoid インストールディレクトリがデフォルトですと"/usr/local"が指定されているので, "~/choreonoid/program"に変更されるように CMAKE_INSTALL_PREFIX オプションを指定します.

```
$ cd build
$ cmake -DBUILD_MULTICOPTER_PLUGIN=ON -
DCMAKE_INSTALL_PREFIX=~/.choreonoid/program ..
```

9. CMake により Makefile 一式がビルドディレクトリ内に生成されたと思うので, Choreonoid のビルドを実行します.

-j オプションにより並列ビルドを行うことができます. -j の後の数値は論理コア数を指定します.

```
$ make -j4
```

10. 最後に Choreonoid のインストールを行います. これを実行することにより, CMAKE_INSTALL_PREFIX オプションで指定したディレクトリ内に Choreonoid の実行ファイル一式がインストールされます.

```
$ sudo make install
```

11. ホームディレクトリに移動し, .bashrc を編集します

```
$ cd ~/
$ gedit .bashrc &
```

12. .bashrc を開いたら, 最終行に環境変数 PATH の設定を追加し保存します.

【追加行】

```
export PATH=${PATH}:~/choreonoid/program/bin
```

13. 編集を保存したら, .bashrc の設定を再読み込みします.

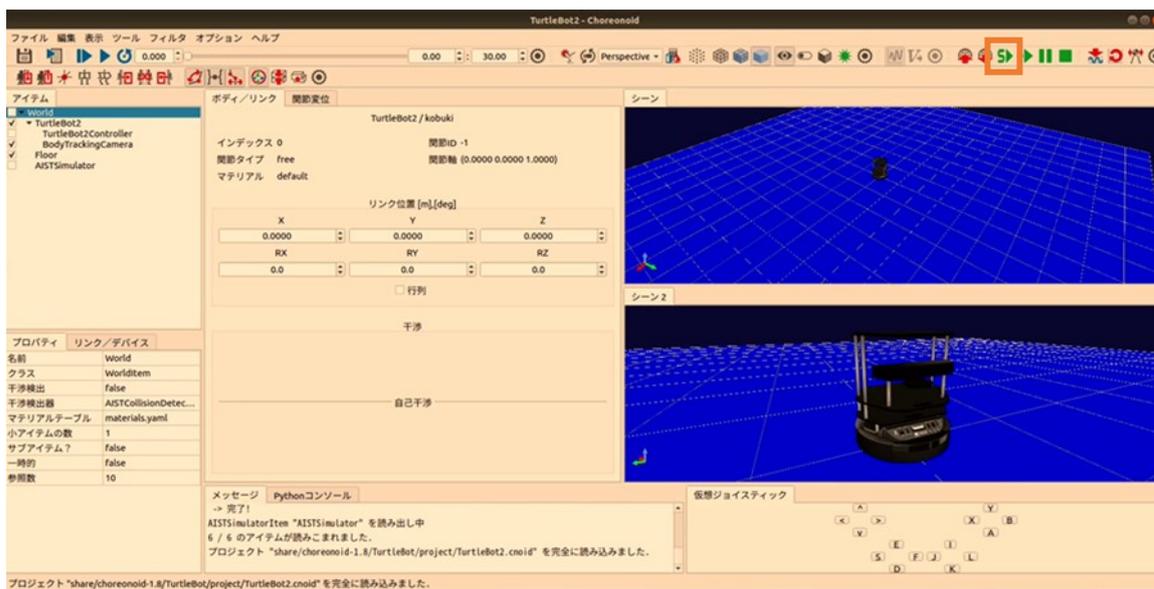
```
$ source .bashrc
```

14. Choreonoid を起動し, TurtleBot のプロジェクトファイルが表示されることを確認してみてください.

```
$ cd ~/choreonoid/program/
```

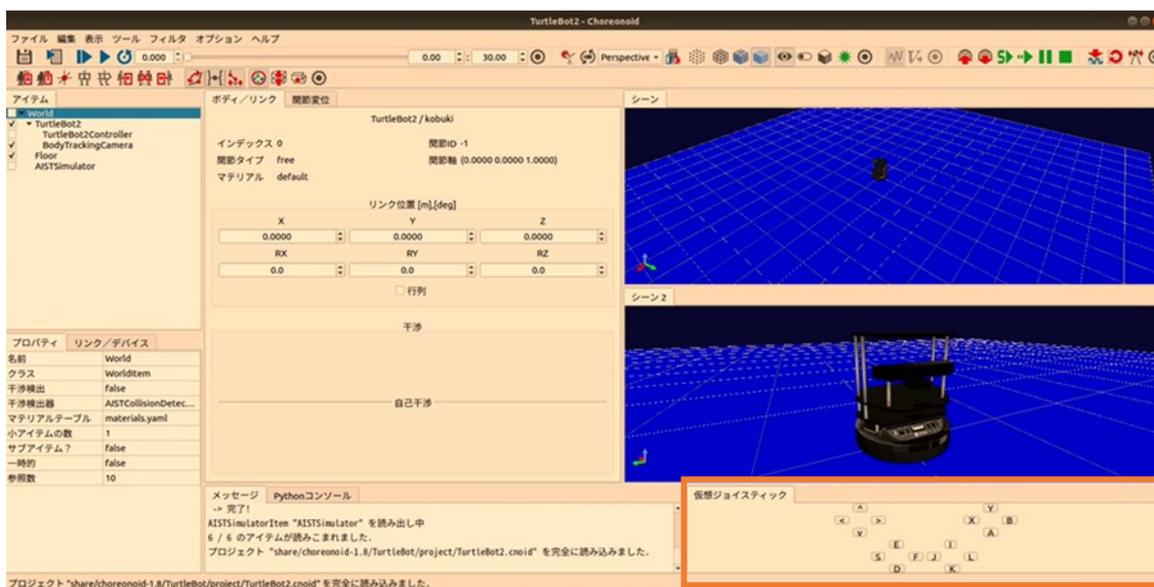
```
$ choreonoid share/choreonoid-1.8/TurtleBot/project/TurtleBot2.cnoid
```

15. プロジェクトが起動したら, シミュレーション開始ボタンを押下します.



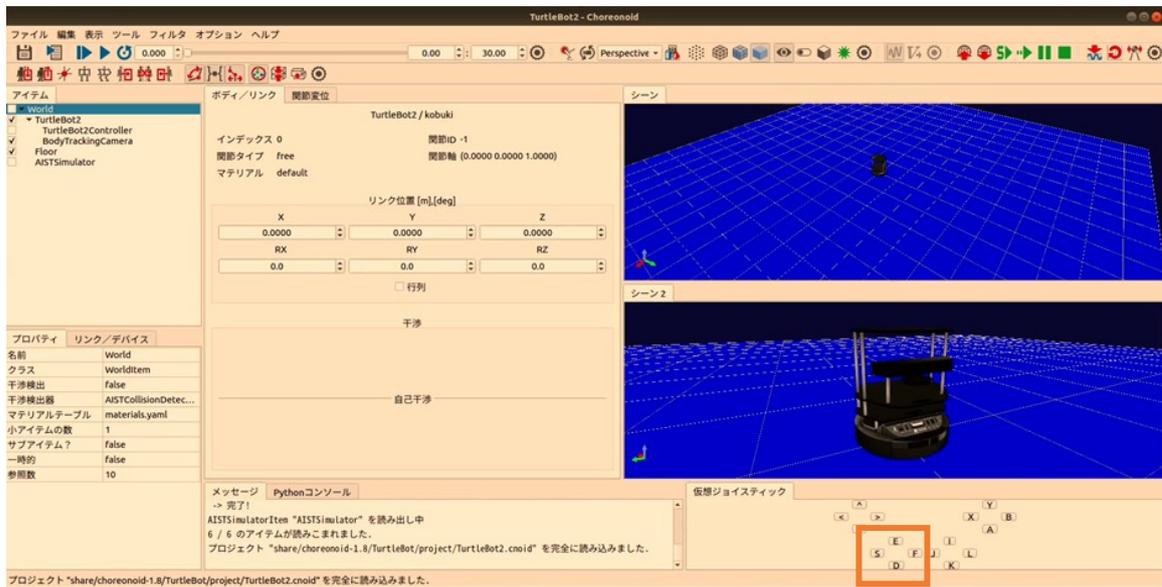
シミュレーションを開始

16. 仮想ジョイスティックビューをマウスでクリックし, アクティブにします.



仮想ジョイスティックをアクティブ

17. キーボードの E, D, S, F いずれかのキーを押下し続けると、前進、後退、旋回の動作をします。



キーボードによる操作