

Sapid インストールマニュアル

– Ubuntu(64bit)[Docker コンテナ] –

愛知県立大学 3年 山本研究室

山田 裕太

2017年6月12日

1 インストール環境

Sapid をインストールした環境，及びパッケージは以下の通りである。

表 1 インストール環境

ホスト OS	Windows7 Professional 64bit
コンテナソフトウェア	Docker Toolbox 1.12.2
コンテナ OS	Ubuntu16.04.1
g++	5.4.0
flex	2.6.0
bison	3.0.4
openjdk	8u111-b14-2
ant	1.9.6
tk-dev	8.6.0
tcl-dev	8.6.0
xutils-dev	7.7
libxml2-dev	2.9.3

2 Docker コンテナの導入

まず、Sapid をインストールするためのコンテナを用意しなければならない。

今回は、Docker Toolbox を用いて Docker をインストールし、Ubuntu コンテナを作成する。

2.1 Docker のインストール [1]

1. Docker Toolbox 公式ページ [2] から Docker のインストールファイル (Windows 版)DockerToolbox-1.12.2.exe をダウンロードする。
2. ダウンロードした exe ファイルを実行し、その後はインストーラの指示に従ってインストールする。

2.2 Ubuntu コンテナイメージの取得 [3][4]

以下のコマンドを用いて、Docker Hub サーバ上の Ubuntu コンテナイメージを取得する。

```
$ docker pull ubuntu:<バージョン番号>
```

ちなみに、バージョン番号を記述せず `docker pull ubuntu` とした場合、自動的に最新のイメージが取得される。

3 インストール環境の構築

3.1 パッケージのインストール

Sapid をインストールするために必要なパッケージを `apt-get` コマンドを用いてインストールする。パッケージをインストールするために、以下のコマンドで Ubuntu コンテナを起動・ログインする。

```
$ docker run -i -t ubuntu /bin/bash
```

以下のコマンドを実行しアップデートする。

```
# apt-get update
```

アップデートが終わったら、以下のようにコマンドを実行しパッケージをインストールする。

```
# apt-get install g++
# apt-get install flex
# apt-get install bison
# apt-get install openjdk-8-jdk
# apt-get install ant
# apt-get install tk-dev
# apt-get install tcl-dev
# apt-get install xutils-dev
# apt-get install libxml2-dev
```

3.2 JAVA_HOME の環境設定

~/bashrc に以下のように JAVA_HOME のパスを追加する。なお、Ubuntu コンテナ内では CUI 環境のみ使用できるので、vim や emacs などの CUI エディタを用いて編集する。

```
JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-amd64
export JAVA_HOME

PATH=$JAVA_HOME/bin:$PATH
export PATH
```

その後、以下のコマンドを実行し、~/bashrc を読み込む。

```
# source ~/.bashrc
```

4 Sapid のインストール

4.1 Sapid のソースのダウンロード及び解凍 [3]

1. ホスト OS 上で最新版の Sapid のソースファイル (Sapid-6.104.4.tar.gz) をダウンロードする .
2. 1Sapid のソースのダウンロード及び解凍 [3]Item.3 のソースファイルを , 前章でコンテナとしてインストールした仮想 Ubuntu に , Docker コマンドを用いて移動する . なお , ホスト OS 上のターミナルで , 以下のようなコマンドを実行することで , 行うことができる . [5]

```
$ docker cp <ソースファイルのパス> <Ubuntu コンテナ ID またはコンテナ名>:<Ubuntu 上の保存先パス>
```

上のコマンドでのコンテナ ID は ,

```
$ docker run -i -t -d ubuntu /bin/bash
```

を実行することで Ubuntu をバックグラウンド起動し ,

```
$ docker ps
```

で現在起動中のコンテナのリストを表示することで確認できる .

3. Ubuntu コンテナ上で Sapid のソースファイルを解凍する . まず , 以下のコマンドでバックグラウンド起動中の Ubuntu コンテナにログインする .

```
$ docker attach <コンテナ ID>
```

次に , Ubuntu 上のソースファイル保存先まで移動し , 以下のコマンドを実行する .

```
# tar xzvf Sapid-6.104.4.tar.gz
```

以上で Sapid のソースのダウンロード及び解凍が終了した .

4.2 tcl,tk のバージョンへの対応

インストールした tcl,tk のバージョンは双方共に 8.6 である . しかし現在の Sapid のバージョンは 8.5 に対応するように設定してあるためエラーが発生する . したがって , ファイル内の記述を変更しなければならない . 解凍したフォルダを <SAPID_TOP> とし , <SAPID_TOP>/Sapid/SapidSite.def を以下のように変更する .

変更前

```
173: /* #define SapidTclIncDir /usr/local/include */
174: /* #define SapidTkIncDir /usr/local/include */
175: /* #define SapidTclLibDir /usr/local/lib */
176: /* #define SapidTkLibDir /usr/local/lib */
177: /* #define SapidTclVersion 8.0jp */
178: /* #define SapidTkVersion 8.0jp */
```

変更後

```
173: #define SapidTclIncDir /usr/include/tcl8.6
174: #define SapidTkIncDir /usr/include/tk8.6
175: #define SapidTclLibDir /usr/lib/tcl8.6
176: #define SapidTkLibDir /usr/lib/tk8.6
177: #define SapidTclVersion 8.6
178: #define SapidTkVersion 8.6
```

変更後,<SAPID_TOP>に移動して次のコマンドを入力する。なお、XMKMF_LOG, MAKE_LOG,INSTALL_LOG がカレントディレクトリ内に生成されるので、それを閲覧することでそれぞれのコマンドに対するログを確認できる。

```
# xmkmf -a 2>&1 | tee XMKMF_LOG
# make 2>&1 | tee MAKE_LOG
# make install 2>&1 | tee INSTALL_LOG
```

最後のコマンドを実行した後、

```
Sapid-6.104.4 ("/usr/local/Sapid"): Install finished.
```

と表示されれば、Sapid のインストールは成功である。

最後に、 ~/.bachrc に以下の記述を追加し、source コマンドで読み込んでおく。

```
if [ -f /usr/local/Sapid/lib/SetUp.sh ]; then
. /usr/local/Sapid/lib/SetUp.sh
fi
```

5 テスト

テストを行う前に, 以下のコマンドでホームディレクトリに `/usr/local/Sapid/sample/test` をコピーしておく.

```
# cp -r /usr/local/Sapid/sample/test ~/Test
```

5.1 dhystone-2.1

まずは `dhystone` のテストに必要な `sutbs-32.h` をインストールする.

```
# apt-get install libc6-dev-i386
```

`~/Test/dhystone-2.1` に移動し以下のコマンドを実行しテストを行う. なお `mkSapid_log` `test_log` がカレントディレクトリ内に生成されるので, それを閲覧することでそれぞれのコマンドに対するログを確認できる.

```
# mkSapid -a 2>&1 | tee mkSapid_log  
# make test-all 2>&1 | tee test_log
```

2 つ目のコマンドを実行した後, `Sapid: Test: dhry: Passed.` と表示されれば, テストは成功である.

5.2 java-zip

`~/Test/java-zip` に移動し, `dhystone-2.1` と同様に, 上記のコマンドを実行しテストを行う. `dhystone-2.1` のテストと同様に, `Sapid: Test: java-zip: Passed.` と表示されれば, テストは成功である.

参考文献

- [1] Docker Toolbox の使い方, <<http://devcenter.magellanic-clouds.com/learning/docker-toolbox.html>>, 2016 年 12 月 15 日
- [2] Docker Toolbox 公式ページ, <<https://www.docker.com/products/docker-toolbox>>, 2016 年 12 月 15 日
- [3] docker コマンド チートシート, <<http://qiita.com/voluntas/items/68c1fd04dd3d507d4083>>, 2016 年 12 月 15 日
- [4] Docker イメージの入手, <<https://thinkit.co.jp/story/2015/09/01/6377>>, 2016 年 12 月 15 日
- [5] Docker cp コマンドでホスト・コンテナ間のファイルのコピーをする, <<http://unskilled.site/docker-cp%E3%82%B3%E3%83%9E%E3%83%B3%E3%83%89%E3%81%A7%E3%83%9B%E3%82%B9%E3%83%88%E3%83%BB%E3%82%B3%E3%83%B3%E3%83%86%E3%83%8A%E9%96%93%E3%81%AE%E3%83%95%E3%82%A1%E3%82%A4%E3%83%AB%E3%81%AE%E3%82%B3/>>, 2016 年 12 月 15 日