Marimba
Installation Guide

Noe Hernandez
no.hernan@gmail.com, no_hernan@ciencias.unam.mx

April 29, 2015

Marimba is a HASKELL program that uses the *gchi* compiler; hence, we have to install the HASKELL platform 2014.2.0.0. Furthermore, at one point of Marimba's *Courcoubetis.hs* module, a system of linear equations needs to be solved. Thus, we need the *linearEqSolver* library that is dated as to Aug 28, 2014 (available from `http://github.com/LeventErkok/linearEqSolver`), which in turn finds the solution by applying the *Z3* theorem prover 4.3.2. We strongly recommend running Marimba on Emacs 24.3 with the HASKELL mode configured.

1 Marimba's installation process in ubuntu 14.04

To download and install the HASKELL platform type the following in a terminal

```
sudo apt-get update
sudo apt-get install haskell-platform
```

Now, the source code for the *Z3* theorem prover has to be downloaded from `https://github.com/Z3Prover/z3`. As a result, a compress zip file is obtained. Extract such a zip file. Before you proceed, make sure you have the tools for building packages; for instance, you might install these building tools with the instruction `sudo apt-get install build-essential checkinstall`. Open a terminal, move to the extracted zip folder containing the *Z3* tool, and run the commands

```
python scripts/mk_make.py
cd build
make
sudo make install
```

Get Marimba from `https://github.com/nohernan/Marimba`, extract the file `Marimba-master.zip` in your *desktop*, and rename the folder `Marimba-master` as `Marimba`. Next, the linear equation solver library has to be installed. So, in a terminal type the commands

```
cd ~/Desktop/Marimba/linearEqSolver/
cabal update
cabal install linearEqSolver
```

Lastly, move to Marimba's root directory, invoke the *ghci* compiler, and load the *Main.hs* module with the instruction `:l Main.hs`. Start the model checker by calling the function `main`.

1.1 Emacs and its haskell-mode in ubuntu 14.04

To install Emacs 24.3 in ubuntu 14.04, type in a terminal `sudo apt-get install emacs24` . In addition, the HASKELL mode is obtained with the command `sudo apt-get install haskell-mode`.

Marimba can be initialised by opening the module *Main.hs* with Emacs, and pressing the `Haskell` menu and the `Load file` option. Next, invoke the function `main` to start interaction with Marimba.

2 Marimba's installation process in Windows 7

Go to `https://www.haskell.org/platform/windows.html` and, depending on your system, download the HASKELL platform installer for *Windows* 32bit or 64bit. Next, install it by double-clicking on it.

Afterwards, download the zip file with the *Z3* theorem prover from `http://z3.codeplex.com/releases`. Extract this compressed file in `C:\`. Add the `z3\bin` folder to the system path as follows:

1. Open the System Panel by going to Control Panel → System and Security → System.
2. Click on Advanced System Setting.
3. Select the Advanced tab.
4. Click the Environment Variables button.
5. Under System Variables, select Path, and then click Edit.
6. Add the route to the *Z3* binaries, e.g., `C:\z3\bin`.
7. Click OK in the Environment Variables window.
8. Click OK in then System Properties window.

Download and extract Marimba from `https://github.com/nohernan/Marimba`. Rename the folder `Marimba-master` as `Marimba`. To install the linear equation solver library, open a command prompt window, change the current working directory to `Marimba\linearEqSolver`, and type the next commands

```
cabal update
cabal install linearEqSolver
```

Finally, return to Marimba's root directory in the command prompt window, and execute the *ghci* command. Next, type `:l Main.hs`, then `main` to start the model checker.

2.1 Emacs and its haskell-mode in Windows 7

Emacs 24.3 is installed by downloading the file `emacs-24.3-bin-i386.zip` from `http://ftpmirror.gnu.org/emacs/windows/`. Extract such a zip file in `C:\`, for example. You can run Emacs without any additional steps but to incorporate icons in your Start Menu, you should run the program `addpm.exe`, which is usually located into the same *bin* directory with `emacs.exe`. Next, open Emacs by selecting it from the Start Menu, press `M-:`[1], then write `(find-file user-init-file)`. In the file that has just been opened, paste the following content:

```
(require 'package)
(add-to-list 'package-archives
  '("melpa" . "http://melpa.milkbox.net/packages/") t)
(package-initialize)
```

and save it by pressing `C-x C-s`[2]. With the changes saved, they'll be applied next time you open Emacs, and every time thereafter. Alternatively, to apply these changes press `M-x`, then write `eval-buffer` at the prompt, and finally press *Enter*. Now your Emacs knows about the MELPA package repository.

Upgrade your installed packages to the latest version by issuing `M-x package-list-packages`. You will get a list of all packages that you may install from MELPA. Refresh such a list to the latest version by

[1] `M-:` is a combination of keys. In this case, you press the `Alt` key and the colon (`:`) key simultaneously.

[2] This is another combination of keys. `C-x C-s` indicates that the keys `Ctrl` and `x` are pressed at the same time. Afterwards, `Ctrl` and `s` are also pressed simultaneously.

pressing `r`. Press `U` to mark all upgradable packages. Finally, to install those packages press `x`. Emacs will ask for confirmation, and will then download and install the new versions and remove the previous ones. Since some of these changes affect parts of Emacs which can't be modified within a running session, you will want to restart Emacs after the package update is complete in order to avoid the random brokenness which may otherwise result.

The following steps define the key bindings to be used, and enable the indentation mode and interaction features:

- Open your personal configuration file. Remember, to do so press `M-:`, then write `(find-file user-init-file)` and finally press *Enter*.
- Add the lines:

```
(add-hook 'haskell-mode-hook 'turn-on-haskell-indentation).

(custom-set-variables
  '(haskell-process-suggest-remove-import-lines t)
  '(haskell-process-auto-import-loaded-modules t)
  '(haskell-process-log t))
(eval-after-load 'haskell-mode '(progn
  (define-key haskell-mode-map (kbd "C-c C-l") 'haskell-process-load-or-reload)
  (define-key haskell-mode-map (kbd "C-c C-z") 'haskell-interactive-switch)
  (define-key haskell-mode-map (kbd "C-c C-n C-t") 'haskell-process-do-type)
  (define-key haskell-mode-map (kbd "C-c C-n C-i") 'haskell-process-do-info)
  (define-key haskell-mode-map (kbd "C-c C-n C-c") 'haskell-process-cabal-build)
  (define-key haskell-mode-map (kbd "C-c C-n c") 'haskell-process-cabal)
  (define-key haskell-mode-map (kbd "SPC") 'haskell-mode-contextual-space)))
(eval-after-load 'haskell-cabal '(progn
  (define-key haskell-cabal-mode-map (kbd "C-c C-z") 'haskell-interactive-switch)
  (define-key haskell-cabal-mode-map (kbd "C-c C-k") 'haskell-interactive-mode-clear)
  (define-key haskell-cabal-mode-map (kbd "C-c C-c") 'haskell-process-cabal-build)
  (define-key haskell-cabal-mode-map (kbd "C-c c") 'haskell-process-cabal)))
```

- Save the file with `C-x C-s`.
- Apply the changes with `M-x eval-buffer` and press *Enter*, or restart Emacs.

Open and load the module *Main.hs* of Marimba by typing `C-x C-f` and writing the path to this module. Next, run `C-c C-l`. You will then be asked about the root of the *Cabal* project as follows:

`Start a new project named ‘‘haskell’’? (y or n)` Press `y` and *Enter*.
`Cabal dir:` (auto-configured value) Press *Enter*.
`Set current directory:` (auto-configured) Press *Enter*.

You will get an interpreter window with a λ prompt. Start Marimba by calling the function `main`.