

# Notice d'installation

1<sup>er</sup> février 2017

## 1 Haskell platform

Il s'agit d'une installation complète de *ghc* (un compilateur de *Haskell*) et de *cabal* (un gestionnaire de bibliothèques *Haskell*).<sup>1</sup>

### 1.1 Machines de l'université

Déjà installé.

### 1.2 Linux Communs (Ubuntu, Debian, Mint, Fedora, Redhat, Gento et Arch)

Présent dans le gestionnaire de paquets standard (du moins dans les versions à jours).

Cas particulier de *Gento* : pour avoir une version plus à jours, vous pouvez utiliser *layman* :<sup>2</sup>

```
$ sudo layman -a haskell
$ sudo emerge haskell-platform
```

Autre cas particulier de *Arch Linux* : Pour une version plus à jours, il est conseillé d'installer les composants séparément, c'est à dire : *ghc cabal-install, haddock, happy* et *alex*.

**Testé sur :** Ubuntu, Debian et Mint

### 1.3 Windows

Installation décrite sur le site officiel.

Téléchargez l'installer (ou bien sa version 32bit).

Après installation, effectuez la commande suivante pour récupérer la localisation du fichier de configuration de *cabal* :

---

1. Un autre choix répandu est l'utilisation de *Stack*, mais de part la nature du projet de programmation, nous vous le déconseillons pour ce cours.

2. Voir le wiki Haskell en cas de soucis.

```
cabal user-config init
```

Puis ajoutez ces lignes a ce fichier de configuration :

```
extra-prog-path: C:\Program Files\Haskell Platform\8.0.1\msys\usr\bin  
extra-lib-dirs: C:\Program Files\Haskell Platform\8.0.1\mingw\lib  
extra-include-dirs: C:\Program Files\Haskell Platform\8.0.1\mingw\include
```

**Non testé** (Test à venir)

## 1.4 Mac OS

Installation décrite sur Le site officiel.  
Téléchargez l'installateur et exécutez le...

**Non testé** (Test à venir)

## 2 Atom

*Atom* est l'*IDE* (environnement de développement) préconisé pour le cours. Il est conseillé de tous utiliser le même ide de part la nature du projet (où vous reprenez le travail d'un de vos camarades...).

### 2.1 Machines de l'université

Déjà installé.

### 2.2 Linux Très Communs (Ubuntu, Debian, Mint)

Présent dans le gestionnaire de paquets standard (du moins dans les versions à jours).

### 2.3 Autres

Description de l'installer à venir, mais disponible sur le site officiel.

## 3 Atom et Haskell

*Atom* est un *IDE* générique, entendez par là qu'il supporte de nombreux langages (comme *emacs* par exemple). Cela veut aussi dire qu'il faut installer les *plugins* correspondant au langage. Pour ce :

1. Lancez *Atom* (tapez **Atom** dans un terminal...).

2. Il va maintenant falloir télécharger les pacages *Atom* relatifs à *Haskell* :  
 Allez dans **Edit->Preferences->Packages**.  
 Les packages “autocomplete-haskell”, “haskell-ghc-mod”, “ide-haskell”, “ide-haskell-cabal” et “language-haskell” sont-ils installés ?  
 Si oui, passez à l’étape suivante, si non vérifiez que vous disposez bien de 25M de libre et téléchargez-les (via **Edit->Preferences-> Updates**).
3. Nous allons maintenant paramétrer Atom et ses packages :  
 Dans **Edit->Preferences->Packages->ghc-mod->Settings**, sélectionnez ‘On Mouse Hover Show’->‘Type and Info’ et ‘On Select Show’->‘Type and Info’.  
 Dans **Edit->Preferences->Packages->ide-haskell->Settings**, cochez l’option “On save Prettify”.  
 Dans **Edit->Keymap**, un fichier “keymap.cson” va s’ouvrir automatiquement, y taper (avant d’enregistrer) :  

```
'atom-text-editor[data-grammar~="haskell"]':
  'ctrl-alt-a': 'haskell-ghc-mod:insert-type'
```

 Cela permet de générer le type automatiquement en tapant “ctrl-alt-a”, d’autres macros sont possibles, voire la doc.

À ce niveau, vous aurez probablement des rapports d’erreurs. Cela est due au fait *Atom* a besoin de bibliothèques *Haskell* spécifiques afin d’afficher les types directement etc.... Pour y remédier, il va falloir télécharger ces bibliothèques via *Cabal*. Le descriptif suivant fonctionne sur des linux standards, mais n’a pas été testé sur Windows et Mac (prévu prochainement) :

1. Vérifiez que *Cabal* est bien installé en tapant `cabal --h` dans le terminal.
2. Tapez
 

```
$ cabal update
```

 puis<sup>3</sup> :
 

```
$ cabal install Cabal
$ cabal install ghc-mod
$ cabal install stylish-haskell
```
3. Relancez Atom.
4. Créez un projet Atom dans lequel vous travaillerez : faites **File -> Add Project Folder...** et choisissez le dossier dans lequel vous allez travailler. Tous les fichiers de ce dossier seront sujet aux features de Atom.

## 4 Utilisation Basic de Haskell et Atom

**Interpreter :** L’interpreteur *Haskell* si lance en tapant simplement *ghci* dans le terminal.

---

3. Comme expliqué dans la doc, il est normalement préférable d’installer ce genre de package dans une *sandbox*, qui est une boite pour les packages permettant la rétro-compatibilité : chaque projet est associé à une sandbox, et si on a besoin d’un package incompatible avec les autres, on crée une nouvelle sandbox pour le projet en question.

**Compiler :** Pour compiler un fichier *Haskell* `exemple.hs`, il faut qu'une fonction `main :: IO ()` soit défini. Dans ce cas tapez simplement dans le terminal :

```
$ ghc exemple.hs
```