

Installation d'OCaml sous Windows

Le but de ce tutoriel est de présenter la procédure d'installation d'OCaml sous Windows.

1 Avantages et inconvénients

avantages

- accessible hors ligne
- permet l'utilisation de VSCode avec le support syntaxique pour OCaml et la complétion
- installe aussi un environnement Unix (shell, commandes de bases) permettant de développer « comme sous Linux »

inconvénients

- installation relativement lourde (temps de téléchargement et d'installation)
- occupation du disque (un peu moins de 2Go entre le système Cygwin, OCaml et VSCode)

On suppose pour ce tutoriel une version de Windows 10 64 bits, correctement mise à jour. L'installation devrait aussi fonctionner sans problème avec Windows 11 mais n'a pas été testée. Les versions inférieures de Windows devraient aussi fonctionner mais certains problèmes d'incompatibilité (versions de DLL) peuvent survenir. Windows sous ARM (Windows sur Microsoft Surface et certains ChromeBooks) n'est pas supporté.

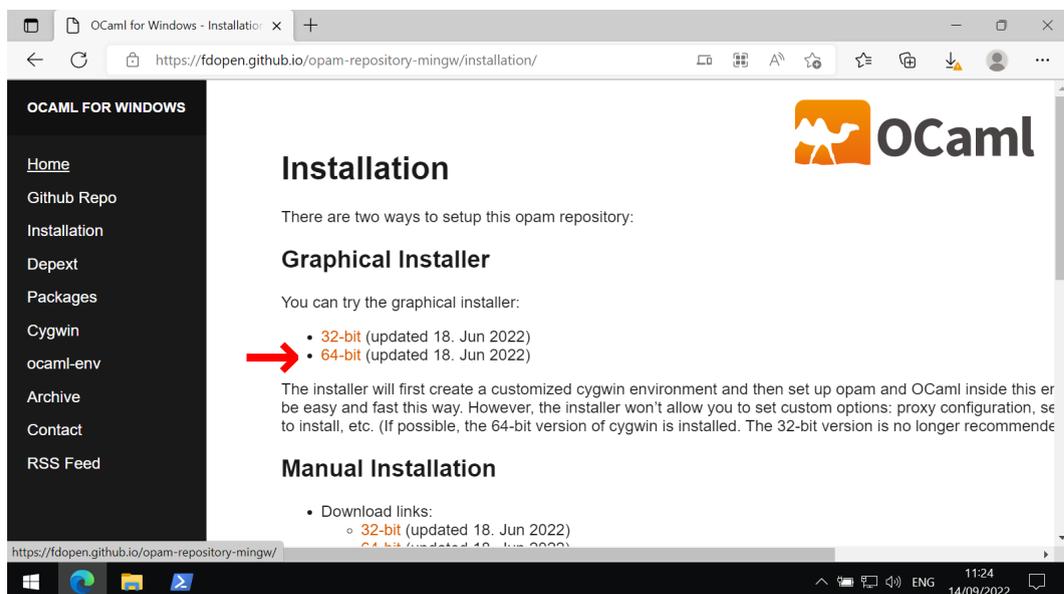
2 Téléchargement et installation d'OCaml for Windows

1. Se connecter sur

<https://fdopen.github.io/opam-repository-mingw/installation/>

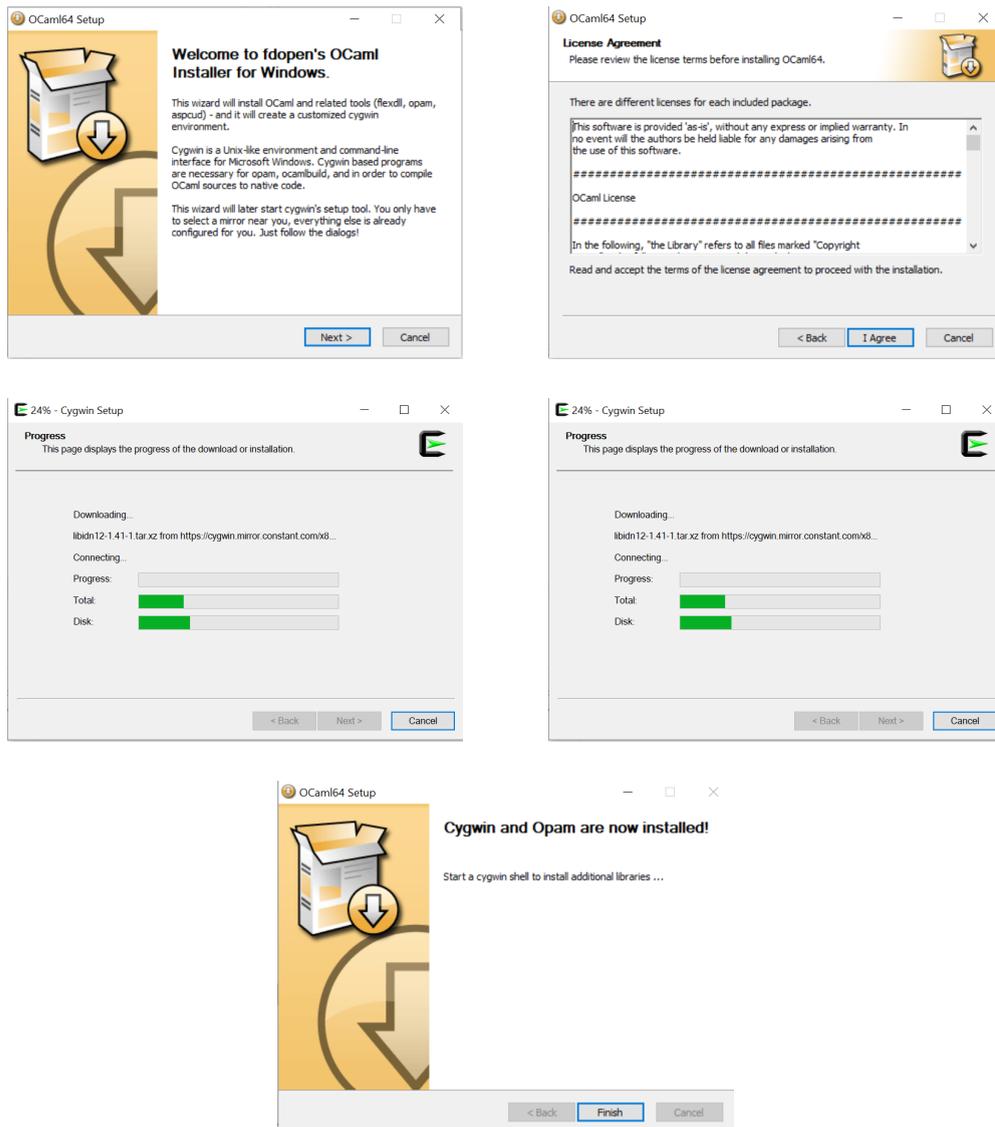
Ce site contient un installeur graphique pour OCaml qui installe en même temps l'environnement Cygwin. Ce dernier permet d'émuler un système Unix sous Windows (Terminal, utilitaires de base tels que `bash`, `grep`, `sed`, ...).

2. Cliquer sur le lien **64-bit** dans la première section (Graphical Installer)



3. Choisir d'accepter le téléchargement et lancer l'installeur téléchargé (fichier `OCaml64.exe`).

4. Avancer dans l'installateur en acceptant et en conservant les choix par défaut ¹.



L'installation peut prendre un certain temps, en fonction de la vitesse de votre connexion réseau et des performances générales de votre machine.

5. Une fois l'installation terminée, vous trouverez deux icônes de terminal sur le bureau. Vous pouvez utiliser n'importe laquelle des deux dans la suite (on mentionnera alors simplement « ouvrir un terminal Cygwin »).

6. Ouvrir un terminal Cygwin. Dans ce terminal :

— le répertoire de l'utilisateur Unix `toto` (chemin `/home/toto`) correspond au chemin Windows :

`C:\OCaml64\home\toto`

De plus, les partitions Windows (`C:`, `D:`, ...) sont accessibles depuis le Terminal Cygwin dans le chemin

`/cygdrive/c, /cygdrive/d,`

Pour rappel, sous Windows, le dossier personnel de l'utilisateur `toto` se trouve par défaut sous :

`C:\Users\toto`

ce dernier contient ensuite les différents dossiers (`Documents`, `Téléchargement`, ...). On peut donc accéder aux fichiers se trouvant dans le répertoire `Documents` de l'utilisateur `toto` depuis le Terminal Cygwin via le chemin :

`/cygdrive/c/Users/toto/Documents`

3 Installation de paquets additionnels pour OCaml

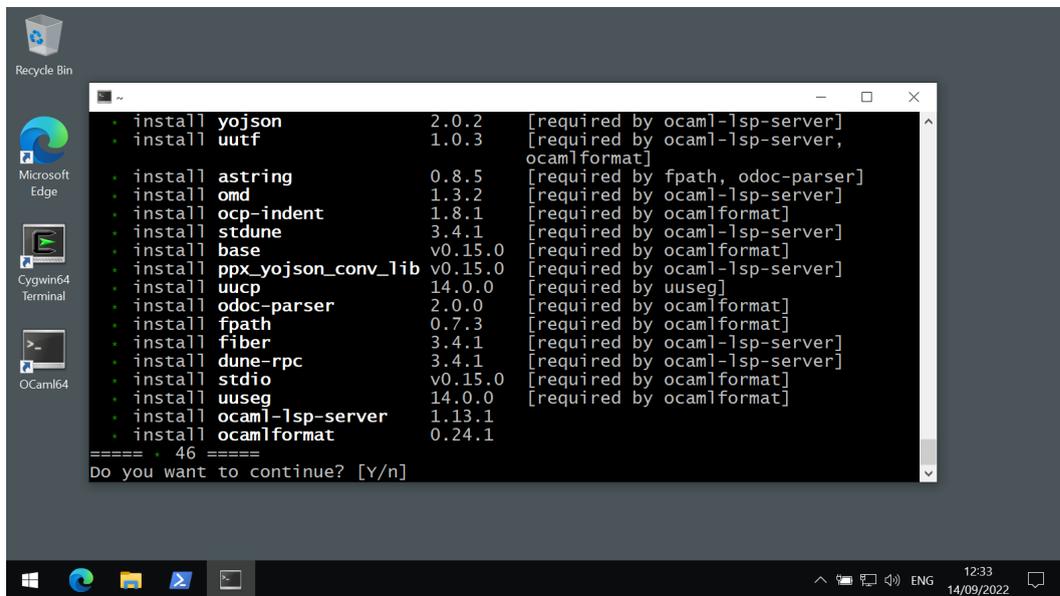
1. Ouvrir un Terminal Cygwin

1. si vous choisissez de modifier le chemin d'installation, en particulier pour choisir une autre partition que `C:`, il faudra vous adapter ensuite

2. exécuter la commande

```
opam install ocaml-lsp-server ocamlformat
```

Valider la liste des paquets à installer avec Y



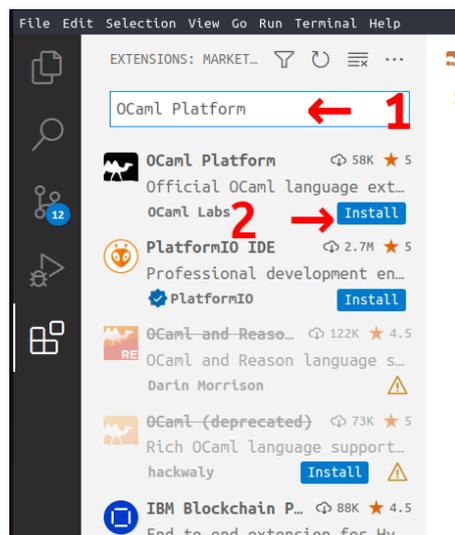
Les paquets mentionnés vont être téléchargés et compilés (ce qui peut prendre un peu de temps)

4 Installation de VSCode et configuration du plugin OCaml

1. Télécharger et installer Visual Studio Code :

<https://code.visualstudio.com/>

2. Une fois installé, lancer VSCode et cliquer sur l'icône des extensions  puis saisir le texte OCaml Platform dans le champs de texte (1). L'extension apparait dans la liste, cliquer sur le bouton pour l'installer (2).



3. Quitter Visual Studio Code

5 Utilisation de VSCode + OCaml

Le plus simple est de travailler exactement comme en TP :

- Ouvrir un Terminal Cygwin
- Créer un répertoire (ou se déplacer vers le repertoire contenant les fichiers à éditer)

— Lancer VScode avec la commande « `code .` » (ne pas oublier le point).

Il est aussi possible de configurer VScode pour lui donner accès au compilateur OCaml et programmes associés depuis l'environnement graphique. Pour cela :

1. Créer un raccourcis vers VSCode sur le bureau
2. Cliquer bouton droit et « Propriétés »
3. Éditer la cible du raccourcis pour ajouter avant le chemin vers l'exécutable VSCode le chemin vers l'utilitaire `ocaml-env-win.exe` :

```
C:\OCaml64\usr\local\bin\ocaml-env-win.exe "C:\Users\toto\chemin\vers\vsCode.exe"
```

4. Enregistrer les modifications.

Il est maintenant possible de lancer VSCode depuis l'icône et le mode OCaml fonctionnera correctement. Sans cette manipulation, il est nécessaire de passer par le Terminal Cygwin pour lancer VSCode avec le support OCaml.