

Éditeurs de texte

Pierre Senellart



Semaine *Informatique pratique*, septembre 2024

Éditeurs de texte

- La programmation, la rédaction de documents \LaTeX , la rédaction de preuve formelle pour assistants de preuve, le développement de sites Web... reposent sur l'édition de fichier textes

- N'importe quel éditeur de texte fait l'affaire

- Grande variété :

Éditeurs de base nano, pico, gedit, kate, Notepad, TextEdit,
vi. . .

Éditeurs intégrés à une application Web Overleaf

Éditeurs d'un environnement de développement intégré
 \TeX works, \TeX nicCenter, Eclipse, PyCharm,
Visual Studio Code. . .

Éditeurs génériques complexes vim, emacs

Éditeurs de texte

- La programmation, la rédaction de documents \LaTeX , la rédaction de preuve formelle pour assistants de preuve, le développement de sites Web... reposent sur l'édition de fichier textes

- N'importe quel éditeur de texte fait l'affaire

- Grande variété :

Éditeurs de base nano, pico, gedit, kate, Notepad, TextEdit,
vi. . .

Éditeurs intégrés à une application Web Overleaf

Éditeurs d'un environnement de développement intégré
 \TeX works, \TeX nicCenter, Eclipse, PyCharm,
Visual Studio Code. . .

Éditeurs génériques complexes vim, emacs

vim

- **vi** : éditeur historiquement influent sous Unix, développé dans les années 70, qui s'est répandu sur toutes les variantes d'Unix
- **vim** : clone libre de vi datant des années 80, y apportant de nombreuses améliorations, toujours développé
- **nvim** : fork libre de vim datant des années 2010, visant les performances et l'extensibilité
- Basé sur la notion de **mode d'édition** : opérations différentes accessibles quand on écrit du texte (mode *insertion*), quand on sélectionne des morceaux de texte (mode *visuel*), quand on exécute des commandes (mode *commande*) ou le reste du temps (mode *normal*)
- Apparence relativement minimaliste, mais possibilités de configuration et d'extension très nombreuses
- Principalement utilisé au sein d'un terminal, même si des interfaces graphiques (gvim, neovide, etc.) existent

emacs

- Éditeur phare (et libre) du projet GNU (le projet qui a fourni la plupart des utilitaires systèmes de Linux) datant des années 80, toujours développé
- Fait usage de nombreuses commandes claviers (avec les touches CTRL et Meta/Win), dont certaines ont eu beaucoup d'influence dans d'autres contextes (shell, etc.)
- Intègre un interpréteur d'une version du langage fonctionnel Lisp, facilitant le développement de macros, d'extensions
- Approche assez maximaliste : l'éditeur de texte peut être l'environnement au sein duquel d'autres programmes tournent (par exemple, un logiciel de courrier électronique)
- En général lancé dans un environnement graphique, mais également utilisable à l'intérieur d'un terminal

Visual Studio Code

- Logiciel propriétaire développé par Microsoft, disponible pour Windows, Mac OS et Linux, basé en très grande partie sur un logiciel libre
- Date des années 2010, évolution des environnements propriétaires spécifiques à certains langages que Microsoft proposait auparavant
- Logiciel moderne, configurable, extensible, devenu très populaire parmi les développeurs
- Revers de la médaille : nettement plus lourd que vim ou emacs
- Uniquement disponible sous forme d'une interface graphique, fait pour être contrôlé à la souris (mais nombreux raccourcis claviers)

Tous proposent

- Coloration syntaxique
- Indentation automatique
- Fonctionnalités avancées de recherche, remplacement, manipulation de texte (en particulier basées sur des expressions rationnelles)
- Correction orthographique à la volée
- Intégration avec des outils implémentant le *Language Server Protocol* pour différents langages, permettant de vérifier la syntaxe du code, de fournir de la complétion automatique sémantique, etc.
- Intégration avec outils externes (compilateurs, systèmes de contrôle de version, etc.)
- Possibilité d'implémenter des extensions (typiquement en VIMScript pour vim, en Lua pour neovim, en Lisp pour Emacs, en TypeScript pour VS Code)

Prenez le temps de choisir !

- Un bon éditeur de texte vous fera **gagner du temps**, **éviter des erreurs** (d'orthographe, de syntaxe, etc.)
- Important de trouver un **confort** dans son éditeur, au sein duquel vous passerez une bonne partie de vos études et de votre carrière
- Dommage de se contenter d'un mauvais éditeur, juste par flemme d'en maîtriser un meilleur
- Un logiciel graphique comme VS Code peut être plus facile d'accès que vim ou emacs, mais ça ne veut pas dire qu'il est le meilleur pour vous