Configuration et utilisation IDE

# Introduction

Un IDE (Environnement de Développement Intégré) est un programme regroupant un ensemble d'outils pour le développement de logiciels :

* éditeur de code avec coloration syntaxique, indentation et complétion automatique ;
* assistant à la conception d'interfaces graphiques, de bases de données, modélisation UML et génération automatique de code ;
* aide à la saisie (bulles d'aides pour saisir les paramètres de fonctions par exemple) et documentation intégrée de l'API mise à la disposition par les bibliothèques du framework ;
* compilateur ou interpréteur et débogueur ;
* refactorisation (changement d'un nom d'identificateur avec correction automatique dans l'ensemble du programme par exemple) ;
* système de gestion de versions, logiciel de suivi de problèmes (issue tracking system), gestionnaire des tâches, outils de gestion de projets, outils collaboratifs ...

Un IDE facilite le développement, mais n’est pas indispensable ; certains développeurs préfèrent installer uniquement le SDK ou utiliser des services en ligne.

Les différents outils (compilateur, système de gestion de versions, …) peuvent être intégrés, mais sont souvent externes. Par exemple, Eclipse est un projet visant à développer un environnement de développement libre, extensible, universel et polyvalent.

Un IDE peut être dédié à un langage de programmation ou en supporter plusieurs : cf Code::Blocks, Eclipse, IDEA, Netbeans, Visual Studio, …

L'objectif de cette activité est d'appendre à installer, configurer et utiliser un IDE, pour le développement en langage Haxe et la bibliothèque OpenFL …

# Installation et configuration d'un IDE

Un IDE seul ne peut transformer un code source en code exécutable : il s'appuie pour cela sur un cadre d'applications (framework ou SDK – Software Development Kit) : celui d'Android (s'appuyant lui même sur celui de Java), celui de Microsoft .net (mono sous Linux), celui de Haxe, …

Les IDEs généralistes (comme IDEA, Eclipse, Visual Studio Code) requièrent par ailleurs l'installation d'une extension (addon) pour « chaque » cadre d'application utilisé. Ils proposent parfois l’installation automatisée du cadre d’applications (effectuer une l’installation manuelle si ce n’est pas le cas).

*Choisir un IDE supportant le langage Haxe (cf* [*http://haxe.org/documentation/introduction/editors-and-ides.html*](http://haxe.org/documentation/introduction/editors-and-ides.html)*), l’installer et le paramétrer sur votre ordinateur personnel (Windows ou Linux).*

# Exemple d’installation manuelle d’un cadre d’applications

* Installer Haxe (et Neko – inclus dans l'installateur Haxe pour Windows) : cf [http://www.haxe.org](http://www.haxe.org/)
* Installer OpenFL à partir du gestionnaire de bibliothèques Haxe « haxelib » : cf [http://www.openfl.org](http://www.openfl.org/download/)
* Sous Windows, si besoin, ajouter à la variable PATH (propriétés systèmes avancées – variables d'environnement) les dossiers contenant les fichiers neko.exe et haxe.exe.

*À ce stade, le SDK est installé, et utilisable pour compiler les programmes (en ligne de commande) et en utilisant un éditeur de texte (Notepad++, SciTE ou Geany par exemple).*