

## LOGICIELS DE GESTION DE VERSIONS DE SOURCE:

Logiciels	Avantages	Inconvénients
<b>Git</b> ( <i>Modèle décentralisé, Open source</i> ): Se voulant simple et performant, dont la principale tâche est de gérer l'évolution du contenu d'une arborescence. Utilise un système de connexion à pair.	<ul style="list-style-type: none"> <li>- Rapide et efficace</li> <li>- Fourni en standard avec une interface web</li> <li>- Dépôt local pour chaque développeur</li> </ul>	<ul style="list-style-type: none"> <li>- Prise en main</li> </ul>
<b>Mercurial</b> ( <i>Modèle décentralisé, Open source</i> ): Écrit principalement en Python. Il a été créé pour s'utiliser via des lignes de commandes et possède un interface web intégré.	<ul style="list-style-type: none"> <li>- Possède son propre interface web</li> <li>- Aucune maintenance requise</li> <li>- Simple et rapide d'utilisation</li> </ul>	<ul style="list-style-type: none"> <li>- Interface rugueux</li> </ul>
<b>Bazaar</b> ( <i>Modèle décentralisé, Open source</i> ): Écrit en Python, réputé pour être rapide et facile d'utilisation.	<ul style="list-style-type: none"> <li>- Intuitif</li> <li>- Bonne documentation</li> <li>- Modèle de branches simplifié</li> </ul>	<ul style="list-style-type: none"> <li>- Tendance à être lent</li> </ul>
<b>Subversion</b> ( <i>Modèle centralisé, Open source</i> ): Inspiré de CVS. Équipé d'une base de donnée sous-jacente capable de gérer les transactions atomiques et permet la gestion de fichiers ou répertoires sans en perdre l'historique.	<ul style="list-style-type: none"> <li>- Maintient complet de l'historique</li> <li>- Simplicité d'utilisation</li> <li>- Toujours supporté malgré son ancienneté</li> </ul>	<ul style="list-style-type: none"> <li>- Pas de commit en offline</li> </ul>
<b>IBM Rational ClearCase</b> ( <i>Modèle centralisé, Prioritaire</i> ): Structurées en deux parties, une contenant les outils de bases et une autre contenant de nouveaux concepts orientés objets.	<ul style="list-style-type: none"> <li>- Mécanisme de gestion de vie d'un logiciel</li> <li>- Possibilité de travailler en multi-site</li> <li>- Sécurité IP efficace</li> </ul>	<ul style="list-style-type: none"> <li>- Payant</li> </ul>
<b>Darcs</b> ( <i>Modèle décentralisé, Open source</i> ): Écrit en langage Haskell, il permet notamment l'échange de correctifs entre différents dépôts sans avoir à créer de correctif supplémentaire officialisant l'échange.	<ul style="list-style-type: none"> <li>- Prend bien en charge les correctifs</li> <li>- Possède moins de commandes</li> <li>- Plus interactif</li> </ul>	<ul style="list-style-type: none"> <li>- Fiabilité</li> </ul>
<b>GNU RCS</b>	<ul style="list-style-type: none"> <li>- Interface utilisateur conviviale</li> <li>- Récupération des données rapide</li> <li>- Optimal pour la consultation régulière d'un même fichier</li> </ul>	<ul style="list-style-type: none"> <li>- Inadapté aux projets d'envergure</li> </ul>

<b>GNU Arch</b>	<ul style="list-style-type: none"> <li>- Facile à administrer</li> <li>- Suit l'ensemble de l'arborescence</li> <li>- Associe un «changeset» à chaque modification</li> </ul>	- N'est plus maintenu
<b>WinDev</b>	<ul style="list-style-type: none"> <li>- Mises à jour prises en charge</li> <li>- Peut lire nativement les informations d'autres logiciels</li> <li>- Prend en charge de nombreux SGBD</li> </ul>	- Utilise son propre langage de programmation
<b>Perforce</b>	<ul style="list-style-type: none"> <li>- Intègre la gestion de déclencheurs</li> <li>- Permet la création de tâches à accomplir sur projet</li> <li>- Gestion des branches intuitives</li> </ul>	- Interface démodé