



## « Formation GIT » GESTION DE VERSION



<b><u>Objectifs</u></b>	<ul style="list-style-type: none"><li>▶ Acquérir les notions de bases en<ul style="list-style-type: none"><li>○ gestionnaires existant</li><li>○ administration de dépôt</li><li>○ branches</li></ul></li><li>▶ A la fin de la formation, les stagiaires seront capables avec GIT :<ul style="list-style-type: none"><li>○ d'utiliser un gestionnaire de version</li><li>○ de travailler avec un dépôt local ou distant</li></ul></li></ul>
<b><u>Public concerné</u></b>	Personnel scientifique, ingénieurs, techniciens doctorants, post-doctorants ayant des besoins de gestion de versions de code.
<b><u>Contenu</u></b>	<p><u>Introduction</u></p> <ul style="list-style-type: none"><li>• Les gestionnaires de version existants (cvs, svn, mercurial, git)</li><li>• Git et ses avantages</li><li>• Fonctionnalités et commandes de base</li></ul> <p><u>Avant de commencer</u></p> <ul style="list-style-type: none"><li>• Installation et configuration du client Git</li></ul> <p><u>Les commandes de base</u></p> <ul style="list-style-type: none"><li>• Créer un dépôt local</li><li>• Ajouter et versionner ces fichiers</li><li>• Supprimer, déplacer et renommer des fichiers</li><li>• Analyser l'historique des révisions</li><li>• Obtenir des informations sur l'état de la copie de travail</li><li>• Ignorer des fichiers</li></ul> <p><u>Branches</u></p> <ul style="list-style-type: none"><li>• Création et gestion des branches</li></ul> <p><u>Collaboration, branches et dépôts distants</u></p> <ul style="list-style-type: none"><li>• Un ou plusieurs dépôts</li><li>• La commande git remote</li><li>• Stockage des identifiants de connexion</li><li>• Gestion des branches distantes</li><li>• Gestion de la synchronisation</li></ul> <p><u>Fusion et conflit</u></p> <ul style="list-style-type: none"><li>• Git merge</li><li>• Git rebase</li><li>• Merge avec conflit</li></ul> <p><u>Bien gérer ses commits</u></p> <ul style="list-style-type: none"><li>• Donner du sens au message de commit</li><li>• Mettre de côté un travail en cours</li><li>• Git add interactif</li></ul>



	<p><u>Réécrire l'histoire</u></p> <ul style="list-style-type: none"> <li>• Les implications de la modification d'historique</li> <li>• Correction / revert de commits et/ou de merge</li> <li>• Réorganiser ses commits</li> </ul> <p><u>Les principaux workflow Git</u></p> <ul style="list-style-type: none"> <li>• Centralized Workflow</li> <li>• Feature Branch Workflow</li> <li>• Gitflow Workflow</li> <li>• Forking Workflow</li> </ul> <p><u>Les plate-formes d'hébergement de dépôts Git</u></p> <ul style="list-style-type: none"> <li>• Github</li> <li>• Bitbucket</li> <li>• Gitlab</li> </ul>
<b><u>prérequis</u></b>	Venir avec son ordinateur (environnement linux ou mac os) ou Windows en téléchargeant le module qui sera fourni)
<b><u>Effectif</u></b>	<b>12 personnes</b>
<b><u>Et Durée</u></b>	<b>29 et 30 mars 2018</b>
<b><u>Lieu</u></b>	<b>CNRS – Délégation Nord pas de Calais. 2 rue des canoniers Lille.</b>
<b><u>Organisateur</u></b>	CNRS DR18 Formation permanente / Min2Rien
<b><u>Inscriptions</u></b>	<p><b><u>AVANT LE 20 février 2018</u></b></p> <p>Fiche d'inscription à envoyer à <a href="mailto:dr18.liste.fp@dr18.cnrs.fr">dr18.liste.fp@dr18.cnrs.fr</a> Pierre Silveira - Responsable Formation CNRS DR18 Tel : 03 20 12 36 88</p> <p><u>Pour les agents non CNRS</u>, contacter au préalable votre service formation afin de valider votre positionnement.</p>