

The logo for the COMPIL conference, featuring the word "COMPIL" in a bold, pink, sans-serif font. The text is set against a background of a grid of small, light-colored squares, some of which contain faint, illegible text, suggesting a code editor or a document with code. The logo is tilted slightly to the right.

COMPIL

Gestion de versions

Introduction

Véronique BAUDIN

e-mail: Veronique.Baudin@laas.fr

PLAN

- Définition et principes de base
- Familles d'outils de gestion de versions
- Plan de la présentation
- Mes sources

Définition

- La gestion de versions de code est l'art de gérer les modifications d'informations
- Applicable au-delà de la gestion de versions de code du monde du développement logiciel
 - Gestion des fichiers d'un site web
 - Gestion de l'historique de /etc pour les ingénieurs système UNIX
 - Documentations, supports de cours, articles, ...

Principes de base

- Permettre un travail collaboratif ou coopératif sur un développement de code ou l'écriture de documents
- Donner un droit à l'erreur en permettant de revenir à des versions antérieures
- Disposer de 2 (ou plus) versions de travail (branches)
- Disposer d'un (ou plusieurs) dépôt(s) de référence

Gestion de version ?

Tout le monde en fait

■ Mode artisanal:

- enregistrer sous (save as)
- `cp monsoft monsoft.saved.190208`
- `cp -r maBiblio maBiblio.190208`

■ Mode professionnel (logiciels dédiés):

- Plus cohérents pour l'archivage
- Permettent de comparer et de gérer en parallèle des versions,
- Quelques outils connus:
 - Commerciaux: ClearCase , Synergy/CM , Visual Source Safe , ...
 - Opensource: CVS , Subversion, git , ...

Familles d'outils de gestion de versions

- Version centralisée
 - Un seul dépôt ou repository contenant toutes les versions: seuls des privilégiés déclarés ont le droit de « commiter » les modifications proposées: CVS, SVN
- Version décentralisée
 - Plusieurs dépôts coexistent
 - Nécessité de synchronisation
 - Nécessité pour les utilisateurs de définir des règles de fonctionnement: git, darcs, svk

Plan de la présentation

- La gestion de version: introduction
- Utilisation classique d'un Système de Gestion de Versions
- Les systèmes centralisés et décentralisés de gestion de version
 - Comparatif général
 - git
- ===== PAUSE CAFE vers 15h30 =====
- Les clients
 - Tortoise
 - Eclipse
 - Décentralisé: git
- Marquage (tags), Retour en arrière, Branching/Merging: comparatif CVS/SVN
- Installation/configuration d'un serveur SVN
- Conclusion



Mes sources

- Les nouveaux systèmes de gestion de version
S. Bortzmeyer (jres 2005) <http://2005.jres.org/paper/2.pdf>
- Le projet OMEGA: <http://www.projectomega.org>
- Version control with Subversion <http://svnbook.red-bean.com/>
- Wikipedia: http://fr.wikipedia.org/wiki/Gestion_de_versions
- PLUME : <http://www.projet-plume.org>