

SCM – Software Configuration Management

Outils de gestion de sources

Fabien Coelho
École des mines de Paris
fabien@coelho.net

révision 309 du document

1

Problématique du développement en équipe

COMMUNIQUER

1. des objectifs, des idées, un design, une intention, une spécification
2. des informations, des pratiques, des standards
3. des sources, des versions
4. des documents
5. des problèmes/anomalies rencontrés
6. des annonces
7. des décisions (leadership ?)

2

Outils divers

messagerie listes de diffusions *dev, user, annonce*

plateforme web annonces, diffusions, tâches, bugs

archives historique, indexation et recherche

dépôt de sources partage des sources du projet

forum, irc discussions éventuelles...

3

Développer est un Art

The art of computer programming, Donald E. Knuth

- 1 problème, 10 programmeurs = 10 solutions...
création, composition, choix, style
- choix guidés par des critères esthétiques
indentation, découpage fonctionnel/classe...
nom des variables, des fonctions, des classes...
contenu des commentaires
- un programme est une preuve de théorème : astuce, élégance
- pas réellement compatible avec les brevets

4

Un logiciel est vivant

- mots : sources, héritage
- opère dans un environnement qui évolue
- évolution perpétuelle du logiciel
 - correction de bogues
 - adaptation à de nouveaux besoins

5

Différents modes de développement de logiciels

1. **personnel** collaboration avec soi-même...
retour en arrière parfois nécessaire...
2. **centralisé** collaboration avec d'autres développeurs
interactions, complémentarité, conflits, propositions
3. **distribué** chacun pour soi (mais on partage ?!)
base commune plus collection de patches

6

1. Développement personnel de logiciels

objectif organisation de son travail

problème perte de données, de fonctionnalités

responsabilité pas de comptes à rendre

coordination non

distance non

centralisation oui

partage direct si accès aux mêmes fichiers !

risques effacement, modifications

exemples mes programmes, mes cours...

7

2. Développement collaboratif centralisé de logiciels

objectif création d'une œuvre commune

problème gestion de versions différentes

responsabilité partage, droits selon parties

coordination oui

distance oui (souhaitable)

centralisation oui

partage naturel, à chaque *update*

risques modifications, conflits

exemples apache, postgresql, subversion

8

3. Développement distribué de logiciels

objectif partage de patches sur une base commune

problème gestion d'une collection de patches

responsabilité non

coordination faible

distance oui

centralisation non

partage publi perso (push), récupérationS (fetch)

risques faibles

exemples début d'apache, linux, netfilter

Le développement sans outils

risques de pertes si erreur, pas de retour en arrière

collaboration partage de fichiers : conflits temps réel

copies inefficaces sauvegarde par duplications. . .

identification des modifications ? auteur, date

recherche de l'état à une date donnée ?

versions multiples corrections multiples des bugs communs ?

Gestions de révisions de fichiers

fichiers de données *Software*

- textes, binaires (bof), répertoires, liens. . . propriétés ?
- documentation : préférer sources xml/sgml/tex, éviter doc/ps/pdf
- interfaçage avec des outils de développement, e.g. eclipse

révisions garder l'historique *Configuration*

- auteurs, messages, dates des transformations
- effacement ? renommage ? modifications de structures ?

gestion évolutions divergentes *Management*

- versions successives *release, revision, bugfix*

Partage de fichiers

verrouillage avant modification *lock-modify-unlock*

- accès local ou droit acquis à distance
- empêche les autres d'avancer. . .
- ne pas oublier de déverrouiller ! vol, timeout ?

fusion de modifications *copy-modify-merge*

- modification sur des copies personnelles
- conflits rares, à gérer à la main
- soumission de modifications (patches) acceptées ou refusées

Besoins

trunk version principale de développement

branches évolution divergentes des sources

- e.g. corrections de bugs dans une *release*
- isolement de développements indépendants

tags marquage d'une version particulière

- la nommer, la dater, la retrouver plus tard
- normalement non modifié/modifiable *frozen*

bundles logiciels liés

- sous projets, librairies partagées apr de apache

Logiciels (libres) supports

centralisé local accès physique nécessaire, bof !

SCCS 1980 Source Code Control System

RCS 1982 Revision Control System

CSSC 1998 Compatibly Stupid Source Control (SCCS libre)

centralisé distant avec accès réseau possible

CVS 1993 Concurrent Versions System

SVN 2001 Sub VersioN

distribué fusions croisées, évolutions indépendantes

GNU Arch 2001 GNU Arch

SVK 2003 SubVersion Kit (?), utilise SVN

GIT 2005 GIT pour sources Linux

Modèle centralisé : progression continue RCS, CVS, SVN

SCCS CSSC RCS historiques de fichiers

- fichiers gérés *individuellement* et *localement*
- pas de manipulations globales directes, renommage, déplacement
- gestion de branches possibles, numérotations 1.3.2.4

CVS arborescence d'historiques de fichiers

- partage réseau des données
- problème renommage, effacement de fichiers et répertoires
- branches possibles avec numérotations à la RCS

SVN historique d'une arborescence de fichiers

- partage réseau des données, accès multi protocoles
- *branches* et *tags* par copies, *bundles* par références externes

Interfaces

- commandes shell Unix: *svn cvs rcs sccs git*
- outils graphiques spécifiques
- plugins explorateurs de fichier (TortoiseSVN)
- plugins eclipse (CVS, SVN. . .)
- modes éditeurs : emacs, vi

Nombreux produits

Centralisé protocoles d'accès ? (fs, http...)

free SCCS, RCS, CVS, SVN, Vesta

com MS Visual SourceSafe, IBM Rational ClearCase, Perforce, Serena Dimensions

Distribué protocoles diffusion ? (mail, ftp, http...)

free GNU Arch, ArX, ARCS, Monotone, Git, Bazaar, Codeville, Mercurial, SVK, Aegis, DCVS...

com MKS BitKeeper, Wandisco, TeamWare

17

Politiques de gestion des sources

droits qui peut faire un commit ?

- de tout le monde à quelques uns ?
- responsabilités ? administration des droits ?
- selon les parties du logiciel (bibliothèques, docs, traductions)...
- soumission de patches, revues strictes pour acceptation

périodes de vie d'un logiciel

développement en cours, nouvelles fonctionnalités

feature freeze correction de bugs (alpha, beta, rc)

release/revision supporté, corrections de bugs uniquement

obsolète d'une version

18

Versions proposées

current version en cours de développement, instable...

selon la politique de gestion des sources

snapshot version d'un jour (une nuit)

alpha version de test peu stable (pre-alpha)

beta version de test assez stable

rc *release candidate*, dernières vérifications

release/revision/bugfix en avant !

19

Théorie des numéros à trois chiffres : 1.2.3

release 1.x à 2.0 fonctionnalités majeures, 3 à 5 ans ?

apache changement d'API des modules

postgresql format bases incompatibles

svn dépôts incompatibles

revision x.2.y à x.3.0 fonctionnalités mineures, 1 an ?

apache recompilation des modules

bugfix x.y.0 à x.y.1 résolution de bugs, 1 à 3 mois ?

php hélas... 4.4.6 change version librairie PCRE

20

Releases...

premier chiffre à 0... ne pas trop en attendre !

http 0.9, 1.0, 1.1

second chiffre pour séparer développement/sorties

linux 2.2, (2.3), 2.4, (2.5), 2.6

apache 2.0, (2.1), 2.2

support des anciennes versions limité dans le temps

apache (... 1.2) 1.3, 2.0, 2.2

postgresql (... 7.2.7.3) 7.4, 8.0, 8.1, 8.2

svn (0.9 1.0) 1.1, 1.2, 1.3, 1.4, 1.5

21

marketing pas rationnel, pousse à aller plus vite

Sun Solaris 2.6 2.7 8 9 10...

Sun Java 1.0 1.1 1.2 1.3 1.4 (Java 2) 5.0 6

MS Windows 3 3.1 95 98 NT ME 2000 2003 XP Vista... années ?

lettres et noms plus facile à mémoriser

Oracle 6 7h 8 8i 9i 10g 11g... *datawareHouse, Internet, Grid*

MacOS cheetah puma jaguar panther tiger leopard...

Debian buzz rex bo hamm slink potato woody sarge etch lenny...

Ubuntu Warty Warthog, Hoary Hedgehog, Breezy Badger, Dapper Drake,

Edgy Eft, Feisty Fawn, Gutsy Gibbon, Hardy Heron

T_EX version 3.141592 (2002) Donald E. Knuth, converge vers π !

METAFONT version 2.71828 converge vers e !

22

Mélange patch sécurité et fonctionnalités

- risques sur les applications, donc pas de déploiement ?
- spécialiste : Microsoft
Windows XP juin 2006 : ajoute Windows Genuine Advantage
- mais aussi : Apple (iPhone)
- mais aussi : ports FreeBSD vs stabilité Debian

23

Conclusion

- pas d'outils adapté à tous les modèles de développement
- un bon outil n'implique pas un bon projet, et inversement !
- subversion c'est bien

Au delà des sources

construction logiciel et doc (configure, make)

tests unitaires ou globaux, portabilité

communication interne/externe, diffusion, documents, anomalies

24

Liens

SVN <http://subversion.tigris.org/>

GNU-Arch <http://www.gnu.org/software/gnu-arch/>



SVK <http://svk.elixus.org/>

CVS <http://www.cvshome.org/>



25

RCS <http://www.gnu.org/software/rcs/>

monotone <http://venge.net/monotone/>



git <http://git.or.cz/>



darcs <http://www.darcs.net/>



bazaar-ng <http://www.bazaar-ng.org/>

misc [http://dmoz.org/Computers/Software/Configuration_Management/](http://dmoz.org/Computers/Software/Configuration/_Management/)

26

List of Slides

- 1 SCM – *Software Configuration Management*
- 1 Outils de gestion de sources
- 2 Problématique du développement en équipe
- 3 Outils divers
- 4 Développer est un Art
- 5 Un logiciel est vivant
- 6 Différents modes de développement de logiciels
- 7 1. Développement personnel de logiciels
- 8 2. Développement collaboratif centralisé de logiciels
- 9 3. Développement distribué de logiciels
- 10 Le développement sans outils

- 11 Gestions de révisions de fichiers
- 12 Partage de fichiers
- 13 Besoins
- 14 Logiciels (libres) supports
- 15 Modèle centralisé : progression continue RCS, CVS, SVN
- 16 Interfaces
- 17 Nombreux produits
- 18 Politiques de gestion des sources
- 19 Versions proposées
- 20 Théorie des numéros à trois chiffres : 1.2.3
- 21 Releases...
- 23 Mélange patch sécurité et fonctionnalités
- 24 Conclusion

24 Au delà des sources

25 Liens