

PLAN DE GESTION DE LA CONFIGURATION DU LOGICIEL

TRAVAIL PRÉSENTÉ À
MME SOUMAYA CHERKAOUI

DANS LE CADRE DU COURS
GEI450, PROJET DE CONCEPTION DE LOGICIELS

PAR L'ÉQUIPE SOKRATE :
SIMON BÉLANGER
YANNICK BROUSSEAU
NICOLAS HATIER
CATHERINE PROULX
FRANÇOIS TREMBLAY

LE 13 JUIN 2001
UNIVERSITÉ DE SHERBROOKE

Sokrate

Plan de Gestion de Configuration du Logiciel (PGCL)

Ce document se conforme à la structure établie par le standard IEEE 828-1990.

1	Introduction	C - 1
1.1	Acronymes	C - 1
2	Gestion	C - 1
2.1	Organisation	C - 1
2.2	Responsabilités	C - 1
2.2.1	Responsable de la Gestion de la Configuration (RGC)	C - 1
2.2.2	Développeurs	C - 1
2.3	Politiques	C - 1
3	Activités	C - 2
3.1	Identification	C - 2
3.1.1	Identification des Élément de Configuration (EC)	C - 2
3.1.2	Convention de noms de fichiers	C - 2
3.1.3	Acquisition d'un EC	C - 2
3.2	Contrôle	C - 3
3.2.1	Demandes de modifications	C - 3
3.2.2	Évaluations des modifications	C - 3
3.2.3	Autorisation des modifications	C - 3
3.2.4	Mise en oeuvre des modifications	C - 3
3.3	Mise en disposition officielle	C - 4
3.4	Audits et revues	C - 4
3.5	Contrôle de l'accès	C - 4
3.6	Sous-traitants et fournisseurs	C - 4
4	Planification	C - 5
5	Ressources	C - 5
6	Mise à jour de la procédure	C - 5
	Annexe I - Procédures CVS	C - 6

1 Introduction

Ce Plan de Gestion de Configuration du Logiciel (PGCL) décrit comment les éléments du projet Sokrate seront gérés.

1.1 Acronymes

GC: Gestion de la Configuration
EC: Élément de Configuration
RGC: Responsable de la GC
SGC: Système de GC

2 Gestion

2.1 Organisation

Un des membres de l'équipe sera désigné comme RGC pour la durée du projet.

2.2 Responsabilités

2.2.1 Responsable de la Gestion de la Configuration (RGC)

Le RGC est responsable de l'organisation et de la gestion de la GC. Il a la responsabilité de maintenir le PGCL. Le RGC est responsable de la mise en place et de l'administration des outils de GC. Il a aussi la responsabilité des copies de sauvegardes. Advenant le cas où les outils utilisés dans le cadre de la GC deviennent non disponibles, le RGC devra développer un plan d'action pour assurer la suite du projet.

2.2.2 Développeurs

Il est de la responsabilité de chaque développeur de se conformer aux règles de GC qui sont publiées dans ce document. Ils doivent aussi s'assurer de connaître adéquatement l'outil utilisé.

2.3 Politiques

2.3.1 Copies conservées

Tous les ECs courants et précédents seront conservés.

2.3.2 Archivage

Un archivage sur CD sera effectué à chaque semaine.

2.3.3 Logiciels utilisés

Le projet Sokrate utilisera un système Concurrent Versions System (CVS) version 1.10 (www.cvshome.org) du côté serveur. Les développeurs pourront utiliser le logiciel client de leur choix, mais le support sera offert pour les logiciels CVS 1.10+ (client) et WinCVS 1.2+ (cvsgui.org).

Le serveur CVS sera situé à l'adresse suivante: scientist.dyndns.org

3 Activités

3.1 Identification

3.1.1 Identification des Élément de Configuration (EC)

Tous les documents ayant rapport au projet sont considérés comme des EC. Le code source, la documentation, les spécifications, les images associées doivent être inclus dans l'outil de GC.

Il n'y a pas de restriction sur les permissions d'ajout de fichier; tous les développeurs peuvent ajouter de nouveaux fichiers au besoin. Chacun doit s'assurer de se conformer à la nomenclature définie au point suivant. Le RGC doit s'assurer que ces règles sont respectés. Advenant une erreur de nomenclature, le RGC devra agir en conséquence et avertir le groupe des modifications apportés au système de GC.

3.1.2 Convention de noms de fichiers

Structure des répertoires

<i>livrables</i>	Contient la documentation du projet ainsi que les fichiers reliés.
<i>code</i>	Contient le code source du projet.
<i>journal</i>	Contient les journaux ("logbook") de chacun des membres de l'équipe.
<i>pv</i>	Contient les compte-rendu des diverses réunion.

Dispositions Générales

- Les noms de fichiers ne doivent pas contenir d'espace ou de caractères accentués.
 - Les extensions de fichiers doivent être écrit en minuscule.
 - Les acronymes doivent être en majuscule.
 - Les noms de fichiers suivent la même nomenclature que pour les noms de variables (voir PAQL)
- Pour le nom des fichiers sources, ils correspondront au nom de la classe contenu sans la lettre préfixant le nom de la classe. Les extensions pour les fichiers sources sont définies dans le standard de code décrit dans l'annexe 1 du PAQL.

Dispositions Spécifiques

- Le nom des fichiers journaux doit être JournalXX.yyy où XX sont les initiales du développeur.
 - Les comptes rendu doivent être nommer sous la forme suivante: JJ-MM-AAAA.yyy
 - Lorsqu'une images ou un document est incorporé à l'intérieur d'une autre document, l'original doit être sauvegardé en indiquant le nom du fichier incluant l'objet suivit d'un tiret "-" et un nom descriptif pour l'objet inclut.
- Ex.: le document PGPL.doc contient une figure Visio décrivant l'échéancier. La figure est intégré comme EC avec le nom PGPL-Ech.vsd

3.1.3 Acquisition d'un EC

En tout temps, tous les EC sont accessibles en lecture seulement. Si un développeur veut travailler sur un des EC, il doit suivre la procédure décrite au point 3.2.1.

3.2 Contrôle

3.2.1 Demandes de modifications

Dans un souci d'efficacité, il est établi qu'il n'est pas nécessaire d'avoir d'autorisation particulière avant de procéder à une modification. Chaque développeur est responsable de s'assurer que ses modifications vont dans le sens des spécifications établies.

Comme les fichiers sont accessible en lecture seule, il est nécessaire d'utiliser une commande d'édition du SGC. Cette commande permet d'informer le reste du groupe qu'un EC est en cours de modification.

Lors de l'acquisition d'un EC, il est préférable de vérifier si quelqu'un d'autre travaille présentement sur ce EC et de communiquer avec pour vérifier que les modifications n'entreront pas en conflit.

3.2.2 Évaluations des modifications

Il n'y a pas d'évaluation spécifique de chaque modification. Par contre des revues systématiques sont définies dans le PAQL section 6.3.

3.2.3 Autorisation des modifications

Bien qu'il ne soit pas nécessaire d'obtenir d'autorisation particulière avant d'apporter des modifications, il faut s'assurer que son code compile sans avertissement s'il s'agit de code source et que les modifications apportées reflètent les décisions de l'équipe.

3.2.4 Mise en oeuvre des modifications

Tous les développeurs ont les droits leur permettant de faire des "checkin" (voir *Mise en oeuvre des modifications* en annexe). Lors du "checkin", il est nécessaire d'entrer un commentaire significatif décrivant les changements effectués sur le fichier.

Lorsqu'on transfère un fichier dans un autre format (ex.: doc à pdf) il faut que le commentaire spécifie le numéro de version du fichier source. De plus, le numéro de version du fichier destination devra être le même que celui du fichier source.

3.3 Mise en disposition officielle

Un label sera effectué à chaque semaine. Des labels supplémentaires pourront être fait au besoin.

3.4 Audits et revues

À chaque réunion hebdomadaire, le RGC devra faire un rapport verbal de l'état de la GC.

3.5 Contrôle de l'accès

L'accès au système de GC se fait à l'aide d'un logiciel client supportant les version de CVS 1.10+. L'accès doit se faire à l'aide de la méthode "pserver" pour assurer un minimum de sécurité. Voir la section 2.3. La méthode "pserver" est une connexion directe utilisant une authentification par mot de passe.

3.6 Sous-traitants et fournisseurs

Le RGC devra se tenir au courant des nouvelles versions de CVS et des derniers bogues rapportés.

4 Planification

Le **tableau 1** présente la liste des labels hebdomadaires qui seront placés sur les EC.

Date	Nom du label
23 Mai	hebdo1
30 Mai	hebdo2
6 Juin	hebdo3
13 Juin	hebdo4
20 Juin	hebdo5
27 Juin	hebdo6
4 Juillet	hebdo7
11 Juillet	hebdo8
18 Juillet	hebdo9
25 Juillet	hebdo10

Tableau 1 - Liste des labels

5 Ressources

Le temps nécessaire pour l'administration générale du système devrait être d'une heure par semaine. Au cours des premières semaine ce temps doit être augmenté afin de configurer le serveur, automatiser les tâches et former les développeurs à l'outil utilisé.

6 Mise à jour de la procédure

Due à l'importance d'un PGCL stable, toutes les modifications apportées à ce document devront être discutées avec l'ensemble de l'équipe.

Dans le but d'augmenter l'efficacité de l'équipe, il est du ressort du RGC d'améliorer la GC. Pour ce faire, plusieurs tâches peuvent être accomplies:

- L'évaluation de nouvelles interfaces d'accès.
- L'évaluation des outils en cours et leur amélioration.
- L'évaluation de nouveaux outils de GC.
- La révision des procédures courantes si elles s'avèrent non efficaces.

Annexe I - Procédures CVS

Accès au serveur

La méthode de connexion sera du type "pserver". Elle est aussi nommé "passwd server file".
Pour accéder à un serveur CVS, il faut donner la racine du serveur (La variable CVSROOT). Pour le projet Sokrate, le CVSROOT sera: ":pserver:user@scientist.dyndns.org:/home/cvsroot".

Copie des fichiers

Pour accéder aux fichiers du projet il faut extraire le module correspondant à l'aide de la commande "checkout", suivit du nom du module voulu. Le nom du module CVS sera "sokrate".

Ajout d'un fichier

Pour ajouter un fichier il faut utiliser la commande "add *nomdufichier*" et ensuite la commande "commit *nomdufichier*" pour envoyer une première version du document au serveur.

Édition d'un fichier

Pour modifier un EC et obtenir les droits en écriture, il est nécessaire d'utiliser la commande "edit"

Voir la liste des fichiers en cours d'édition

La commande "editors" permet de voir les fichiers qui sont en train d'être modifiés et de connaître la personne qui fait ces modifications.

Mise en oeuvre des modifications

La sauvegarde des modifications aux fichiers vers le serveur CVS se fait à l'aide de la commande "commit". Cette commande équivaut au terme "checkin" utilisé dans plusieurs logiciels semblables. L'ajout du paramètre "-r *nouvelleversion*" permet de forcer la version du fichier soumis à un numéro spécifique.