

OUTILS DE GÉNIE LOGICIEL

Gestion de la configuration logiciel (S. C. M.)

OBJECTIF

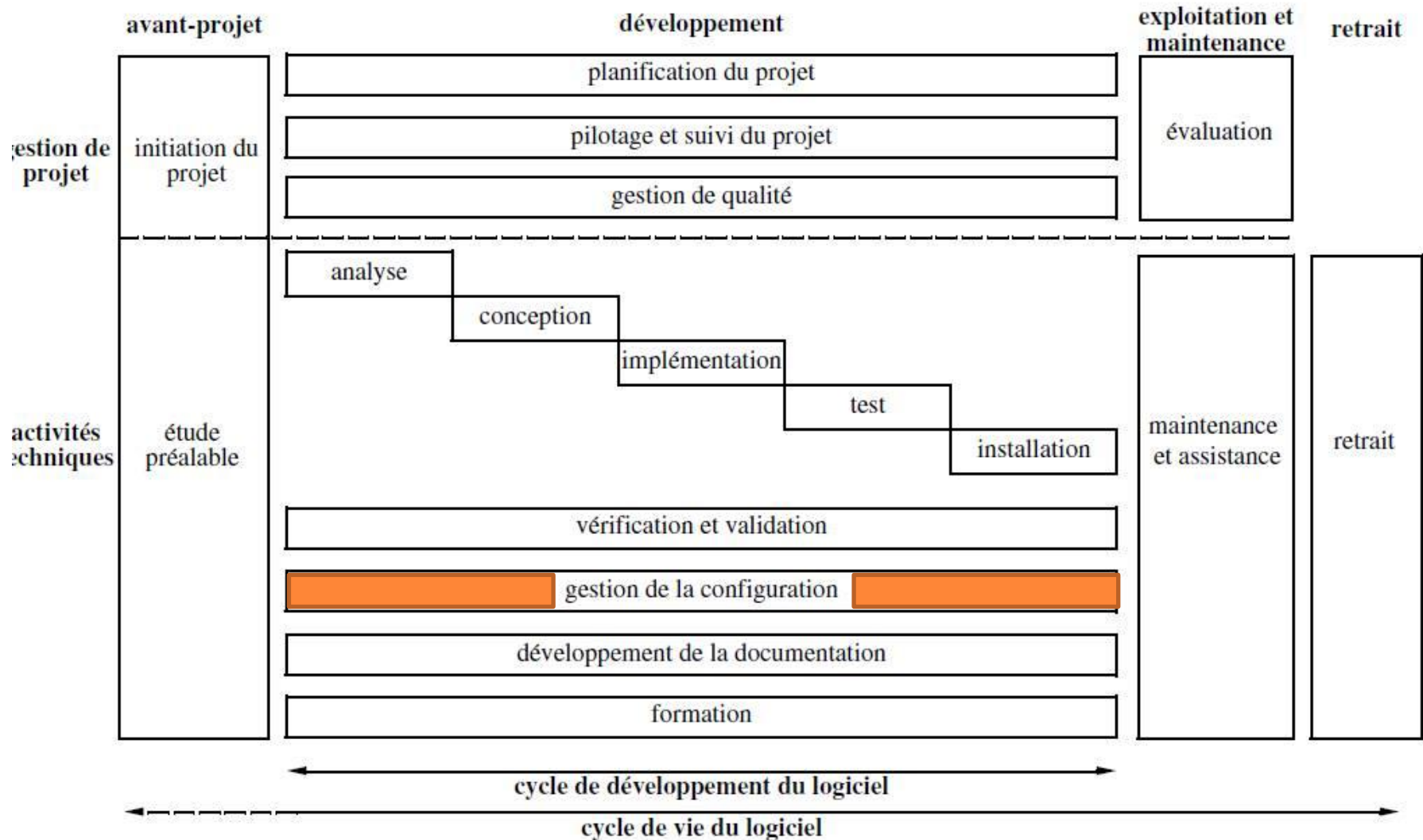
- Comprendre la notion de gestion de configuration (G. C.) et son rôle comme outil de développement logiciel
- Comprendre les concepts liés à la gestion de la configuration
 - La terminologie et les concepts de base
 - Les différentes activités de la G. C. (la SCM)

PLAN

1. Introduction
2. Définition de la gestion de la configuration (GC, SCM)
3. Intérêt de la GC pour le développement logiciel
4. Différentes activités de la GC
5. Concepts de base de la GC
6. Outils de la GC

1. INTRODUCTION

RAPPELS : DIFFÉRENTES ACTIVITÉS AUTOURS DU LOGICIEL



1. INTRODUCTION

- Un système est composé d'un ensemble de composants organisés effectuant une ou plusieurs fonctions
- Il est souvent construit par une équipe de développement (développement team)
- Les composants du système subissent de perpétuels modifications:
 - Au cours des différentes phases de développement
 - Au cours de son exploitation sur le marché avec apparition de nouveaux besoins
- Retombés sur les coûts et les délais ainsi que la qualité du produit livré



Le contrôle de ces aspects relève de

la Gestion de Configuration

2. DÉFINITION : GESTION DE CONFIGURATION LOGICIELLE (G.C.)

- «Software Configuration Management » (SCM)
- La Gestion de Configuration Logiciel (GC) est une activité d'ingénierie logicielle s'occupant de la gestion des changements apportées aux logiciels
- C'est un ensemble de procédés ou méthodologies permettant de gérer, de contrôler et de suivre l'évolution d'un logiciel :
 - Lors des différentes phases de sa construction (en projet).
 - en cours de production ou sur le marché (les livrables)

2. DÉFINITION :

GESTION DE CONFIGURATION LOGICIELLE (G. C.)

- Une discipline où des techniques de gestion et de contrôle sont appliquées pour :
 - L'identification, documentation des différentes caractéristiques de tout les composants du logiciel
 - Le suivie et contrôler de leur modification
 - Reporting du statut des différents composants à tout moment
 - Vérification de la compatibilité des changements au besoins
- Elle couvre le cycle de vie du logiciel:
 - En cours de développement
 - En prototypage
 - Lors de la phase de maintenance

Standard pour le logiciel :

« Plans de gestion de configuration » [IEEE 610]

3. INTÉRÊT DE LA GC (SCM)

- Améliore la fiabilité et la qualité des logiciels en offrant :
 - Un ensemble d'outil pour :
 - identifier et contrôler la documentation, le code, les interfaces
 - Un référentiel support à toutes les phases du cycle de vie (code REPOSITORY)
 - Soutenir le développement par une méthodologie qui répond:
 - Aux exigences des changements
 - À l'évolution des normes
 - Une politique d'organisation
 - Philosophie de gestion des équipes
 - Produire et gérer et les informations sur le produit
 - Le contrôle des changements, des tests, des sorties (release), des audits, etc.

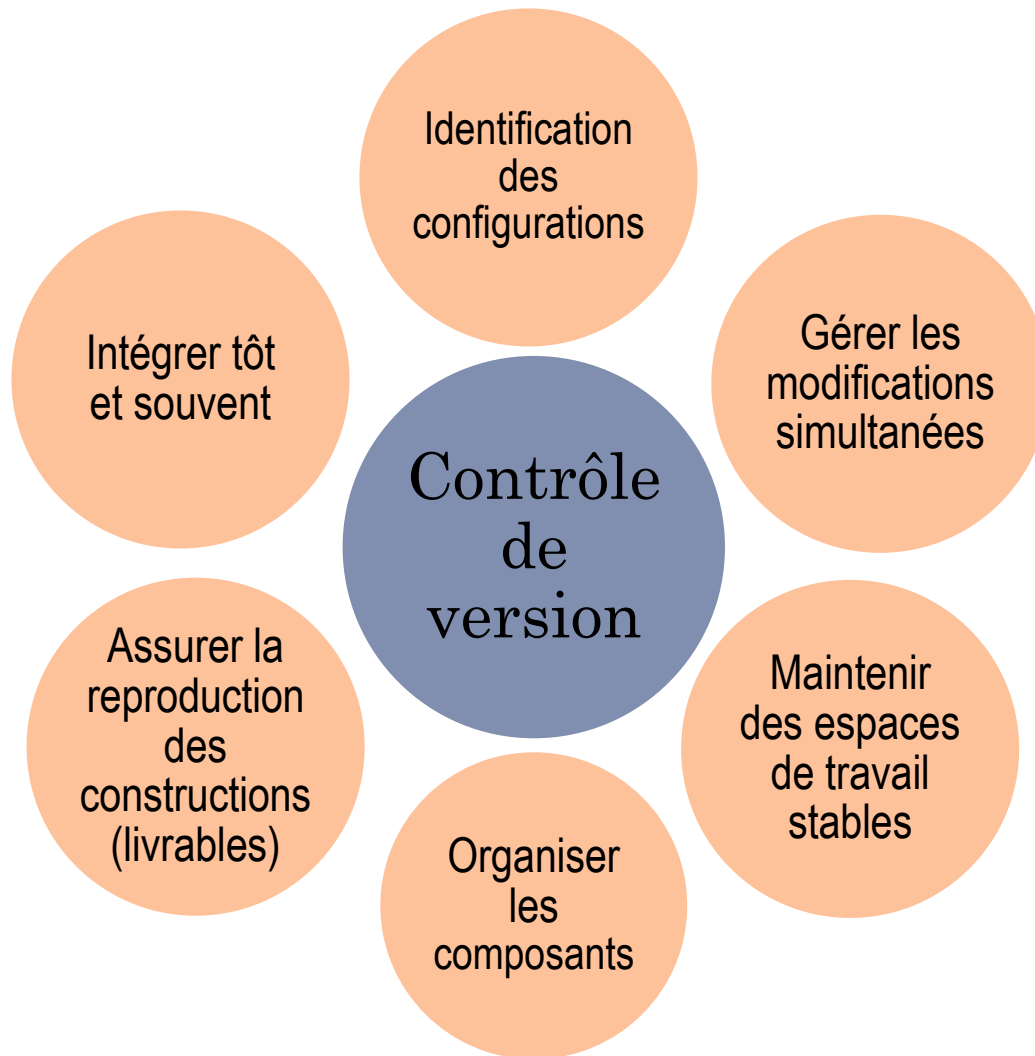
3. INTÉRÊT DE LA GC (SCM)

- Pour le travail d'équipe: assurer aux développeurs un
 - Mécanisme de partager leurs composants et différents éléments et objets logiciels (classe, biblio., fichiers...etc.)
 - Mettre à jour ces objets d'une manière contrôlée
- Pour la maintenance: Les équipes doivent
 - Réaliser des projets dans plusieurs versions
 - Être en mesure de rétablir la version antérieure
 - Fusionner les modifications dans le développement actuel
- Pour l'assurance de stabilité: être en mesure de revenir à des versions antérieures des composants ayant été conformes à un certain niveau de qualité

4. ÉLÉMENTS PRIS EN CHARGE PAR LA G.C.

- Entités et composants qui peuvent constituer les éléments d'une configuration :
 - Les documents de gestion du projet;
 - Les documents techniques de réalisation;
 - Les manuels d'utilisation et d'exploitation;
 - Les programmes sources et les moyens permettant de produire et de reproduire le programme machine;
 - Les jeux de données de tests, y compris leurs résultats, les procédures et scénarios de test

5. ACTIVITÉS LIÉES À LA GC (SCM)



6. CONCEPTS DE BASE DE LA GC

1. La configuration
2. La version (révisions)
3. Le référentiel
4. Ligne de référence (ligne de base) « Baseline »
5. Ligne du code « Code line »

6. CONCEPTS DE BASE DE LA GC :

6.1) LA CONFIGURATION

- Un ensemble d'éléments logiciels ;
 - Composant,
 - Classe,
 - Bibliothèques,
 - Ressources : image, url...etc.)
 - Code source
 - Code exécutable

constituant le logiciel en cours de développement à un instant donné.

- Une configuration est l'assemblage du logiciel pour être utilisé dans une opération relevant de son du cycle de vie
- Exemple :
 - Configuration pour le débogage ou pour les testes
 - Configuration pour le prototypage

6. CONCEPTS DE BASE DE LA GC :

6.1.1) ÉLÉMENT DE LA CONFIGURATION

- Un agrégat du logiciel susceptible de subir une modification au cours du développement composant la configuration :
 - Composant,
 - Classe,
 - Bibliothèques,
 - Ressources : image, url...etc.)
 - Code source
 - Code exécutable
 - Testes
 - Documentation

6. CONCEPTS DE BASE DE LA GC :

6.1.2) RELATION ENTRE ÉLÉMENTS DE LA CONFIGURATION

- La relation entre les éléments de la configuration et entre ces éléments et leur composants
- L'évolution des éléments de configuration engendre l'évolution de la relation qui existe entre eux et celle qui existent entre les éléments et leurs constituants
- Cette relation est aussi un élément de la configuration

6. CONCEPTS DE BASE DE LA GC :

6.2) LA VERSION

- Un état d'un élément de configuration produit par une modification (évolution) au cours du processus de développement
- Une version peut être :
 - Une « révision » : remplacer une autre version
 - Une « variante » : ajouter la nouvelle version sans remplacer une autre
- « **Les révisions se remplacent** / **les variantes coexistent** »
- Une configuration constituée d'un ensemble de versions (révisions ou/et variantes) de ses différents éléments constitue une version de la configuration initiale

6. CONCEPTS DE BASE DE LA GC :

6.3) LE RÉFÉRENTIEL

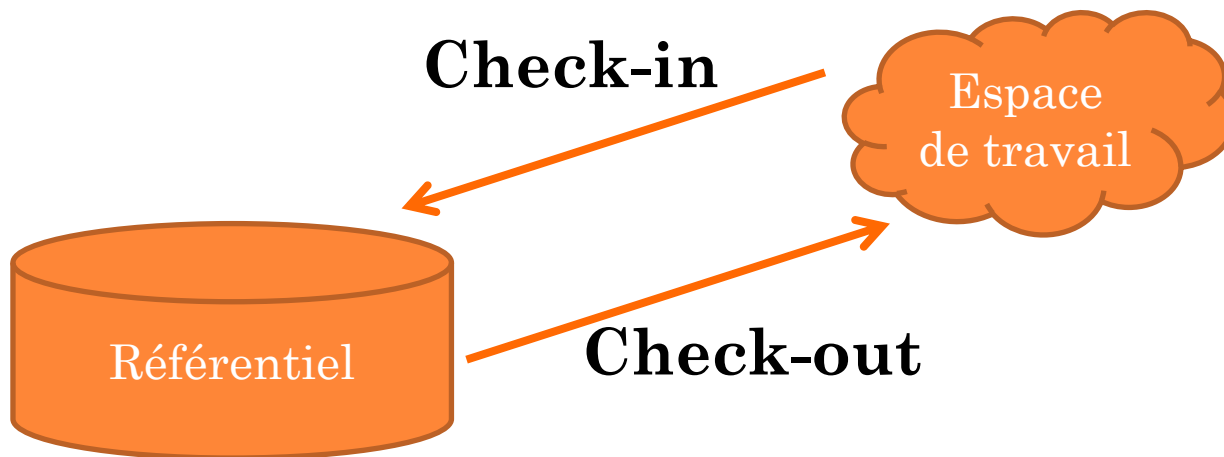
- « Repository » en anglais
- Une structure de stockage des différentes versions des configurations d'un logiciel permettant :
 - le contrôle,
 - le suivi,
 - la production

des différentes livrables du logiciel par les différentes équipes qui interviennent

6. CONCEPTS DE BASE DE LA GC :

6.3.1) OPÉRATIONS DE TRANSFERT

- Deux transferts entre le référentiel et les espaces de travail des développeurs:
 - Dépôt « Check-in » : transfert d'un élément de la configuration de l'espace de travail vers le référentiel
 - Retrait « Check-Out » : transfert d'un éléments de la configuration du référentiel vers l'espace de travail



4. CONCEPTS DE BASE DE LA GC :

4.3.2) ACCÈS CONCURRENTS

- Deux stratégies pour gérer les accès de plusieurs développeurs à un même élément:
 - Pessimiste : verrouillage de l'élément par premier développeur qui l'utilise et le déverrouille à la fin
 - Risque d'oubli de déverrouiller
 - Optimiste : des copies sans créées à chaque retrait et au dépôt on oblige d'abord à retirer la dernière version et fusionner avant de faire dépôt
 - Suivre une ligne de code

6. CONCEPTS DE BASE DE LA GC :

6.4) LIGNE DE RÉFÉRENCE (LIGNE DE BASE)

- « Baseline » en anglais
- Une version de la configuration stable, validée après être formellement examinée et approuvée, sur laquelle le processus de développement se base pour effectuer des modifications futures
- La configuration de départ dans un projet est souvent considérée comme la ligne de référence (de base) qui peut évoluer vers d'autres lignes de références (de bases) au cours du temps durant le processus de développement.

6. CONCEPTS DE BASE DE LA GC :

6.4) LIGNE DE RÉFÉRENCE (LIGNE DE BASE)

- La ligne de référence (de base) ne peut être modifiée qu'en suivant des procédures formelles de contrôle des modifications.
- À un instant donné du processus de développement une configuration dite actuelle est composée de :
 - La ligne de base
 - Les modifications approuvées formellement apportées à cette ligne de base
- Les différentes lignes de bases constituent une traces (historiques) des modifications

6. CONCEPTS DE BASE DE LA GC :

6.5) LIGNE DE CODE

- « Codeline » en anglais
- La ligne d'évolution des différents éléments de la configuration à partir d'une ligne de référence.
- La ligne de code retrace les différentes révisions que subissent les différents éléments d'une configuration (les différents fichiers sources par exemple)
- A un instant donné la ligne de code contient une collection des différentes versions des éléments la composant la configuration.
- Un aperçu (cliché) de la ligne de code constitue une version qui peut être étiqueté

6. CONCEPTS DE BASE DE LA GC :

6.5.1) LABEL D'UNE LIGNE DE CODE

- Une étiquette qui identifie une version de la ligne de code
- La version étiquetée avec un label constitue un point de retour en cas de modifications mal réussie.



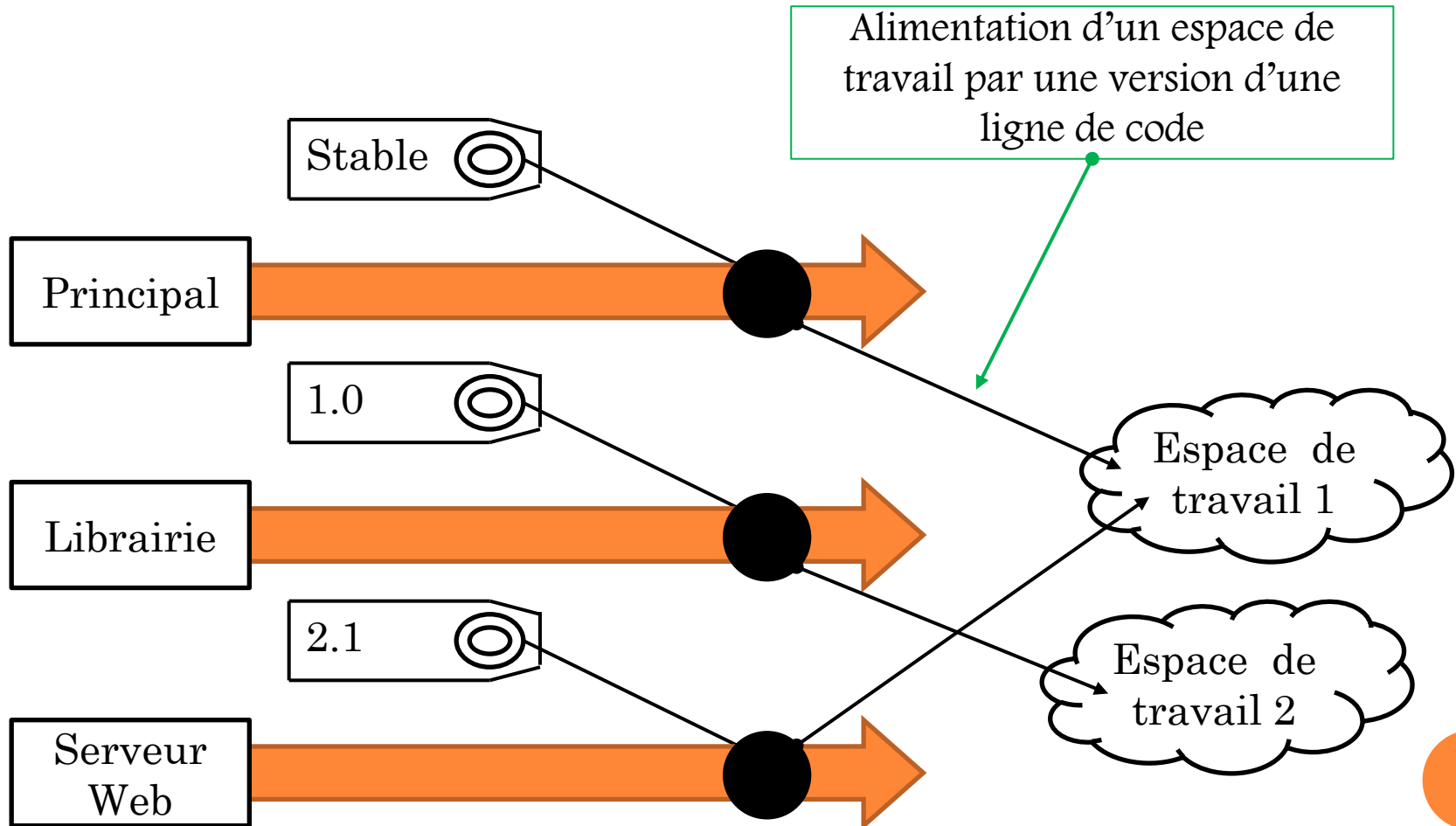
4. CONCEPTS DE BASE DE LA GC :

4.5.3) ESPACE DE TRAVAIL ET LIGNE DE CODE

- Un espace de travail « workspace » constitue l'ensemble des éléments utilisés par le développeur à moment donné
- Un espace de travail est associé à une ou plusieurs lignes de code
- Chaque ligne de code à une politique déterminant son objectif et les conditions des révisions
 - Exemple :
 - Ligne de code de développement des librairies
 - Ligne de code de développement de la GUI
 - Ligne de code principale (le cœur)

6. CONCEPTS DE BASE DE LA GC :

6.5.3) ESPACE DE TRAVAIL ET LIGNE DE CODE



6. CONCEPTS DE BASE DE LA GC :

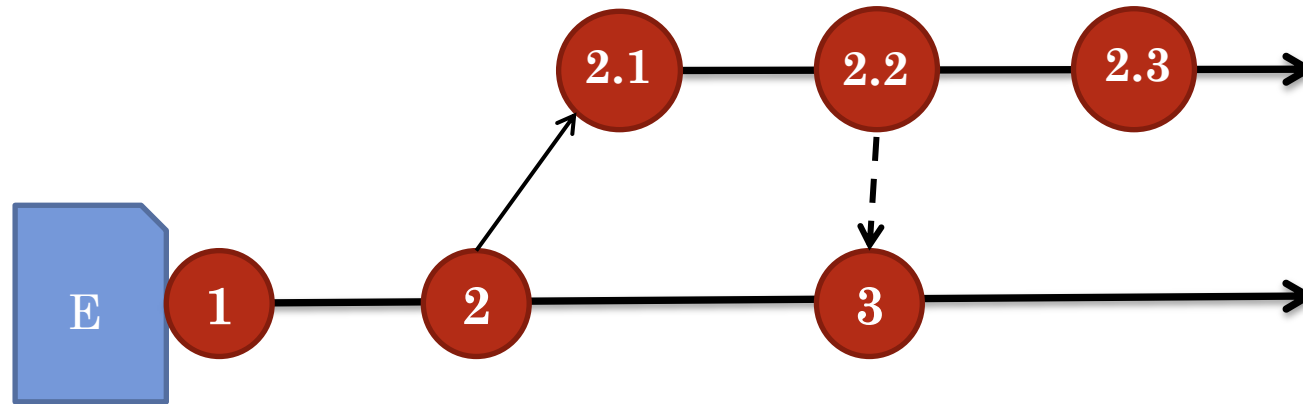
6.5.4) LES BRANCHEMENTS

- Une branche d'un élément de la ligne de code est une variante qui est créée à un instant donné et évolue indépendamment de l'élément d'origine sur une ligne propre à elle
- Une branche peut être tirée à partir d'une ligne de code principal ou d'une ligne de code appartenant à une branche
- Une branche peut donner naissance à une nouvelle ligne de code
- Une branche peut être fusionnée avec la branche mère (le tronc)

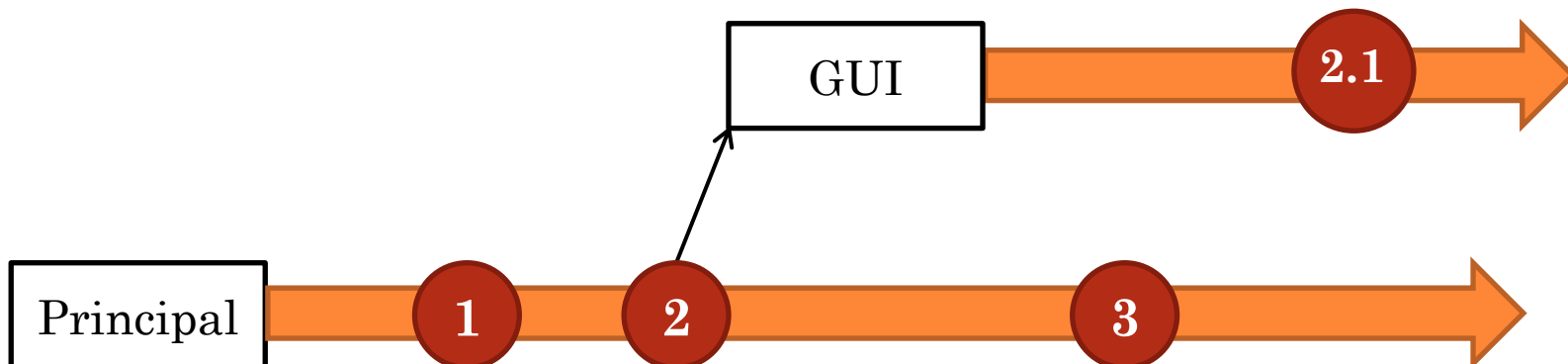
4. CONCEPTS DE BASE DE LA GC :

4.5.4) LES BRANCHEMENTS : EXEMPLE

○ Branchement d'un élément



○ Branchement d'une ligne de code



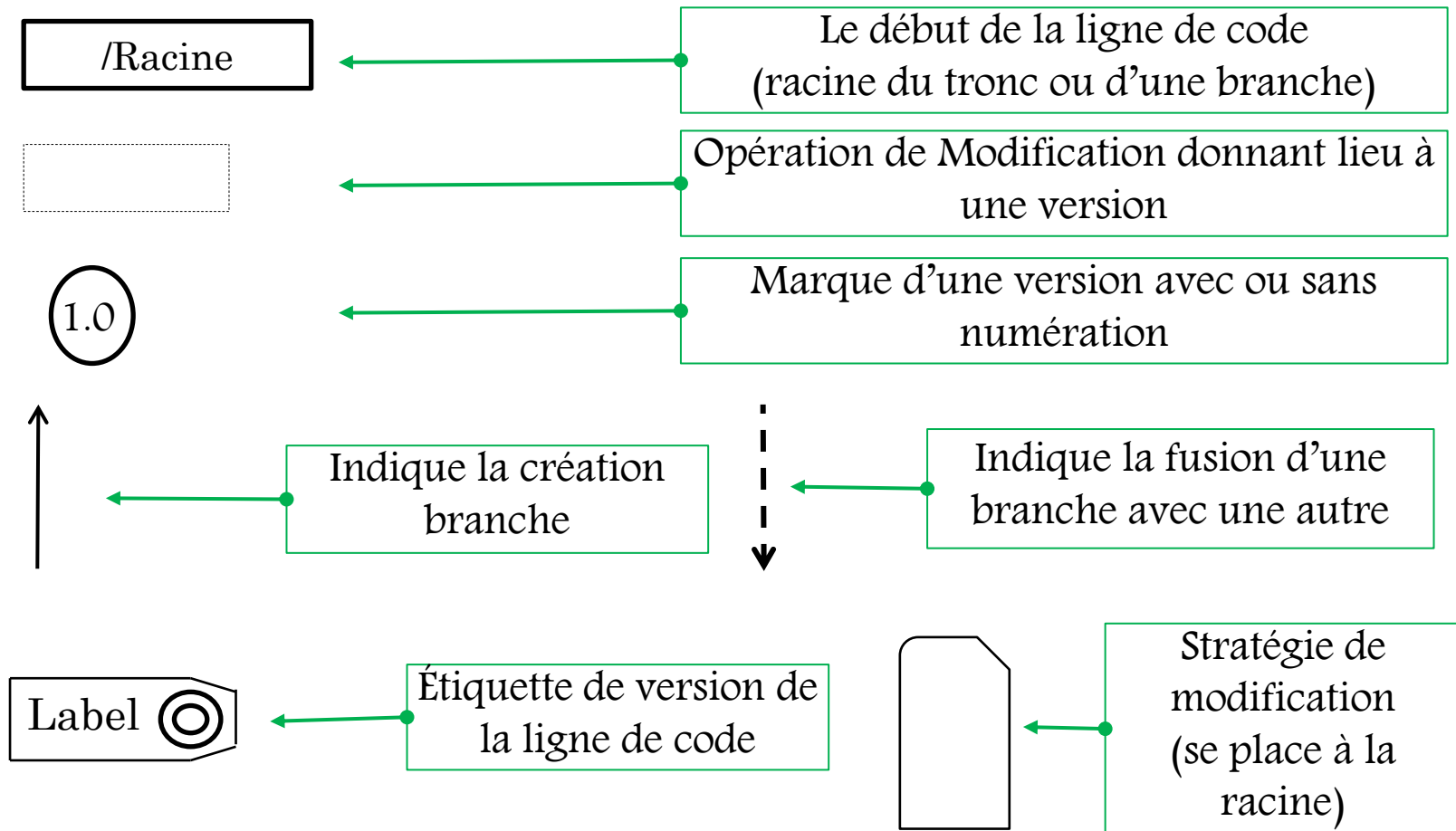
6. CONCEPTS DE BASE DE LA GC :

6.5.5) GRAPHE DE BRANCHEMENT

- C'est un graphe permettant de modéliser les différentes lignes de code dans une configuration donnée en partant d'une ligne de base donnée.
- Il permet de suivre les différentes évolutions que subit la configuration
- Il est composé d'un :
 - Tronc (trunk) représentant la ligne de code principale
 - Branche représentation la création d'une nouvelle lignes de code pour un élément (variante) ou pour toute la configuration
 - Les fusions qui représentent les intégrations des variantes à la version principale

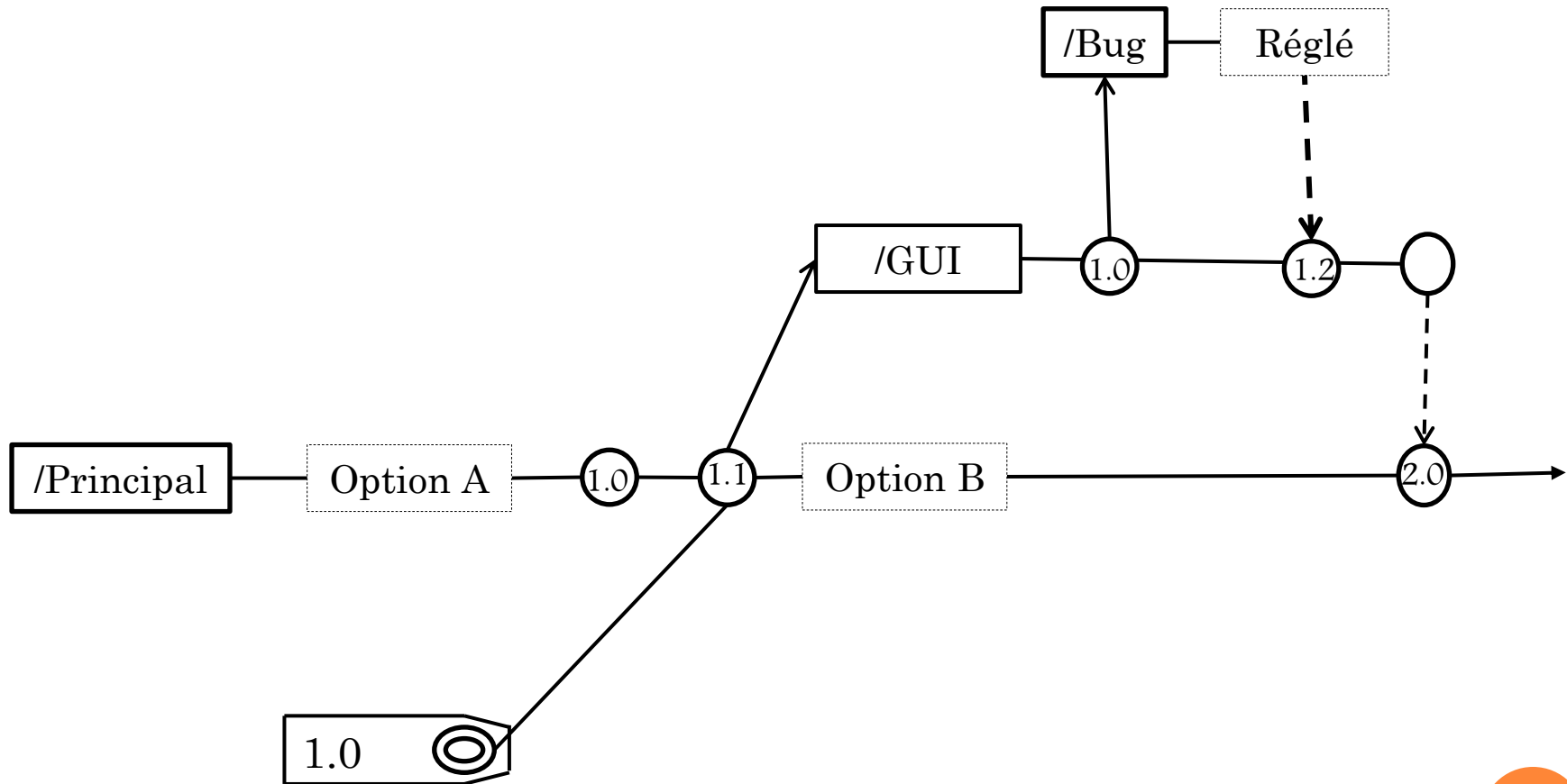
6. CONCEPTS DE BASE DE LA GC :

6.5.5) GRAPHE DE BRANCHEMENT: NOTATION



6. CONCEPTS DE BASE DE LA GC :

6.5.5) GRAPHE DE BRANCHEMENT : EXEMPLE



7. OUTILS DE LA GC

- Maven
- SVN
- Clearcase (IBM)
- Source Safe (Microsoft)
- Vesta (open source)