



# DEPLOIEMENT ET MISE EN SERVICE

M.SENE

# CHAPITRE 1 :CYCLE DE VIE DU LOGICIEL

- Définition
- MODELE DE CYCLE DE VIE
- EXEMPLES PRATIQUES

Deploiement et mise en service/M.SENE

# CHAPITRE 1 :CYCLE DE VIE DU LOGICIEL

## DEFINITION

- Le cycle de vie du logiciel regroupe toutes les étapes de l'existence d'un logiciel, de son apparition à sa disparition.
- Ce cycle de vie se compose de plusieurs étapes afin de séparer les rôles impliqués dans la construction de logiciels.

# CHAPITRE 1 :CYCLE DE VIE DU LOGICIEL

## DEFINITION

- La définition d'un découpage en étapes permet d'isoler les étapes de construction du logiciel, afin de vérifier et valider chacune des étapes indépendamment les unes des autres, et d'assurer une certaine qualité du logiciel tout au long du cycle de vie.
- En effet, plus une erreur est détectée tôt dans le cycle de vie d'un logiciel, plus la correction est aisée à mettre en oeuvre.

# CHAPITRE 1 :CYCLE DE VIE DU LOGICIEL

## DEFINITION

- À l'inverse si une erreur dans les premières étapes du cycle de vie est détectée en fin de cycle, le coût de mise en oeuvre de correction du logiciel devient plus conséquent, puisque la modification du résultat de l'une des étapes du cycle de vie impacte toutes les étapes en aval de celle-ci.

# CHAPITRE 1 :CYCLE DE VIE DU LOGICIEL

## MODELE DE CYCLE DE VIE

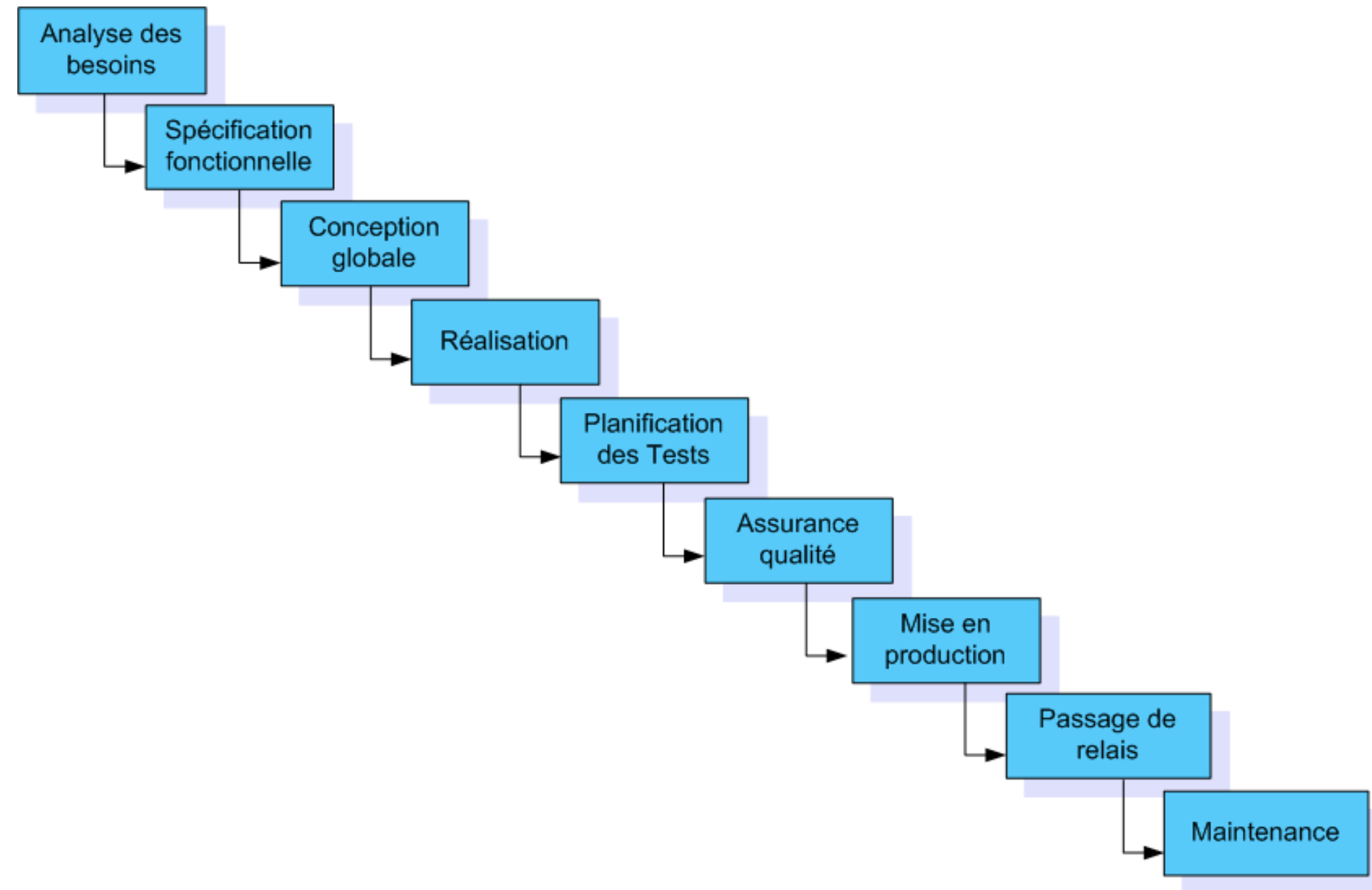
Il existe différents modèles de cycle de vie du logiciel. Parmi ceux-ci, on peut citer

- le modèle en cascade
- le modèle dit en V, ou encore le modèle en spirale de Boehm

# CHAPITRE 1 :CYCLE DE VIE DU LOGICIEL

## MODELE DE CYCLE DE VIE

- Modele en Cascade



Deploiement et mise en service/M.SENE

# CHAPITRE 1 :CYCLE DE VIE DU LOGICIEL

## MODELE DE CYCLE DE VIE

- Modele en Cascade
  - Analyse de besoins : l'étape d'analyse des besoins correspond à la phase d'étude des logiciels existants et des nouveaux besoins du client. Le terme de client désigne une personne ou un organisme qui fait appel aux services de concepteurs en vue d'acquérir un logiciel répondant à l'un de ses besoins.

Deploiement et mise en service/M.SENE



# CHAPITRE 1 :CYCLE DE VIE DU LOGICIEL

## MODELE DE CYCLE DE VIE

- Modele en Cascade
  - La spécification fonctionnelle : elle correspond à l'établissement d'un cahier des charges décrivant les principales fonctionnalités du logiciel, ainsi que l'interface homme-machine (IHM) du logiciel final.

Deploiement et mise en service/M.SENE

# CHAPITRE 1 :CYCLE DE VIE DU LOGICIEL

## MODELE DE CYCLE DE VIE

- Modele en Cascade
  - La conception globale du logiciel : elle consiste en la définition de l'architecture globale du logiciel. Chaque fonctionnalité du logiciel est contenue dans un module et l'interaction entre les différents modules est établie. Cette phase correspond à l'établissement de l'architecture logicielle.

Deploiement et mise en service/M.SENE

# CHAPITRE 1 :CYCLE DE VIE DU LOGICIEL

## MODELE DE CYCLE DE VIE

- Modele en Cascade
  - Réalisation : la phase de réalisation, ou implémentation de chaque module de l'application
  - Enfin, après les tests (unitaires, d'intégration ou de qualification), viennent les phases d'Assurance qualité que l'on nomme aussi Vérification et Validation(V&V).

Deploiement et mise en service/M.SENE

# CHAPITRE 1 :CYCLE DE VIE DU LOGICIEL

## MODELE DE CYCLE DE VIE

- Modele en Cascade
  - La validation est la phase qui consiste à prouver que le logiciel réalisé accomplit bien l'action que l'on attend de lui, tandis que la vérification consiste à prouver que le logiciel a été fait correctement —i.e. qu'il ne produit pas d'erreur non prévue

Deploiement et mise en service/M.SENE

# CHAPITRE 1 :CYCLE DE VIE DU LOGICIEL

## MODELE DE CYCLE DE VIE

- Modele en Cascade
  - l'Assurance qualité, qui consiste à faire en sorte que le logiciel passe tousles tests avec succès,
  - le Passage de relais qui consiste à documenter le logiciel et transférer les compétences d'exploitation au client, et enfin la phase sur laquelle porte notre étude

Deploiement et mise en service/M.SENE

# CHAPITRE 1 :CYCLE DE VIE DU LOGICIEL

## MODELE DE CYCLE DE VIE

- Modele en Cascade
  - la phase de Mise en production. Cette phase regroupe la totalité des opérations qui visent à mettre le logiciel en état de fonctionnement, à le mettre à jour vers de nouvelles versions, ou à adapter son fonctionnement suite à des événements externes au logiciel (dûs à son contexte). Cette phase porte également le nom de Déploiement logiciel.

Deploiement et mise en service/M.SENE

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

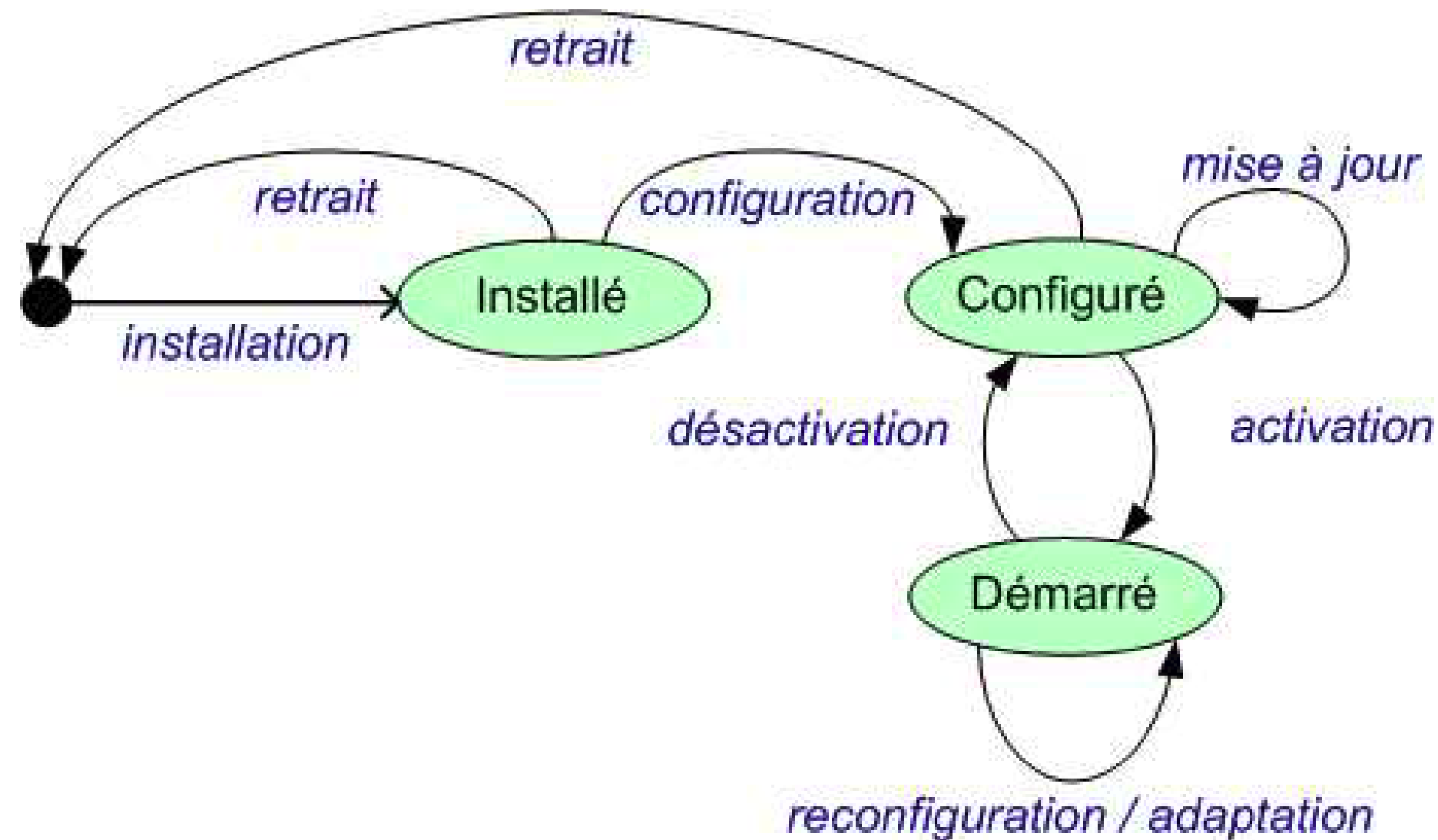
- INTRODUCTION et DEFINITION
- ETAPES CYCLE DE VIE DEPLOIEMENT
- PARAMETRES DU DEPLOIEMENT
- DESCRIPTION DU DEPLOIEMENT
- VERIFICATION DU DEPLOIEMENT

Deploiement et mise en service/M.SENE

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## INTRODUCTION

- un cycle de vie du déploiement comprend différentes étapes, représentées sur la figure suivante





# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## INTRODUCTION

- Ces différentes étapes sont séparées en deux rôles :
  - le rôle de producteur et celui de consommateur de logiciel.
  - Le premier de ces rôles consiste à conditionner un logiciel afin d'en fournir une version livrable, et le second consiste à traiter cette version livrable pour la rendre exécutable pour l'utilisateur.

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## INTRODUCTION

Le déploiement est le fait d'introduire une nouvelle solution/plateforme technique et des services dans une organisation d'une manière coordonnée. Un déploiement réussi repose sur une planification prospective, des ressources adéquates, une surveillance et une évaluation continue et une communication solide.

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## INTRODUCTION

Le déploiement est le fait d'introduire une nouvelle solution/plateforme technique et des services dans une organisation d'une manière coordonnée. Un déploiement réussi repose sur une planification prospective, des ressources adéquates, une surveillance et une évaluation continue et une communication solide.

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## I.ETAPES DU CYCLE DE VIE DU DEPLOIEMENT

1. L'installation est la première étape de ce cycle.

Ce processus est en charge de transférer les paquets du logiciel, c'est-à-dire les unités élémentaires de déploiement pour un logiciel, sur les machines concernées. Ces unités sont fournies par le producteur de logiciel et contiennent toutes les ressources (bibliothèques, exécutables, fichiers de configuration, etc.) du logiciel, ainsi que d'éventuels descripteurs spécifiant les ressources requises pour le déploiement du logiciel.

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## I.ETAPES DU CYCLE DE VIE DU DEPLOIEMENT

2.La configuration est l'étape qui suit directement l'installation.

Durant cette étape, les paramètres du logiciel installé sont fixés en fonction des propriétés de la machine hôte, —i.e. la machine qui héberge ce logiciel. Par exemple, il s'agit de fixer le chemin vers un fichier dans le système de fichiers.

Une fois cette étape achevée, le logiciel est prêt à fonctionner.

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## I.ETAPES DU CYCLE DE VIE DU DEPLOIEMENT

3.L'activation (ou démarrage) est l'étape qui consiste à exécuter le logiciel installé et configuré proprement. Une fois cette étape achevée, le logiciel est en état de fonctionnement. Le logiciel ne changera d'état qu'après l'étape de désactivation (ou arrêt). Cette étape est symétrique au démarrage, —i.e. elle replace le logiciel dans son état après la configuration.

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## I.ETAPES DU CYCLE DE VIE DU DEPLOIEMENT

4.la procédure de mise à jour peut être mise en oeuvre.

Cette étape consiste à faire migrer le logiciel configuré vers une version supérieure. Cette opération nécessite souvent l'arrêt préalable du logiciel, et est effectuée lors de changements réalisés par le producteur du logiciel.

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## I.ETAPES DU CYCLE DE VIE DU DEPLOIEMENT

4.la procédure de mise à jour peut être mise en oeuvre.

Cette étape consiste à faire migrer le logiciel configuré vers une version supérieure. Cette opération nécessite souvent l'arrêt préalable du logiciel, et est effectuée lors de changements réalisés par le producteur du logiciel.



# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## I.ETAPES DU CYCLE DE VIE DU DEPLOIEMENT

5.Le logiciel peut également être soumis à des reconfigurations.

Il s'agit là encore d'une modification du logiciel installé. Mais, contrairement à la mise à jour, la reconfiguration consiste non pas à remplacer le logiciel par une version supérieure, mais à modifier la configuration de la version courante. Qui plus est, ce changement doit pouvoir être opéré sans arrêt du logiciel.

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## I.ETAPES DU CYCLE DE VIE DU DEPLOIEMENT

6.Enfin un logiciel à l'exécution peut être sujet à adaptation. L'adaptation est une forme particulière de reconfiguration, guidée par des changements extérieurs. Ces changements peuvent concerner la machine hôte sur laquelle le logiciel est installé (système de fichiers, ressources système, etc.) ou le logiciel lui-même. L'adaptation est un processus visant à maintenir la cohérence et l'intégrité du logiciel dans le cas de changements éventuels.

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## I.ETAPES DU CYCLE DE VIE DU DEPLOIEMENT

7.Enfin, après l'arrêt du logiciel, la désinstallation du logiciel (ou retrait) peut être appelée.Il s'agit de l'étape symétrique de l'installation, visant à supprimer entièrement le logiciel de la machine hôte.

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## II.PARAMETRE DU DEPLOIEMENT

- Ces différentes étapes du cycle de vie du déploiement ne peuvent raisonnablement pas être appliquées de manière homogène. En effet, différents paramètres entrent en ligne de compte lors de l'exécution de ces tâches. Ces paramètres concernent le logiciel lui-même, mais également la machine hôte sur laquelle celui-ci est déployé.

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## II.PARAMETRE DU DEPLOIEMENT

- Tout d'abord, il est évident que le processus de déploiement de n'importe quel logiciel dépend des caractéristiques de la machine hôte.
- La première caractéristique concerne les capacités matérielles de la machine. Ainsi la puissance de calcul, la capacité de stockage ou encore la connectivité de la machine hôte— i.e. les moyens d'accès au réseau de cette machine— sont des paramètres qui peuvent influencer sur la réalisation du déploiement logiciel.

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## II.PARAMETRE DU DEPLOIEMENT

- En effet, un type de logiciel plus ou moins coûteux en termes de ressources peut ne pas être déployable sur une machine trop peu puissante. De la même manière, le format des commandes à envoyer sur les machines pour déployer un logiciel dépendent du protocole d'accès fourni par la machine concernée. D'autres paramètres à prendre en compte concernant la première couche logicielle de toute machine, c'est-à-dire le système d'exploitation.
- En effet, cette couche, ayant pour but de fournir un accès logiciel aux ressources de calcul de la machine, peut posséder différentes caractéristiques. Tout d'abord le protocole d'accès à distance, indispensable pour agir sur les machines, varie d'une machine à l'autre.

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## II.PARAMETRE DU DEPLOIEMENT

- Il existe différents protocoles pour l'accès à distance aux machines, parmi ceux-ci on trouve le Secure SHell (SSH), le TErминаl NETwork (Telnet). Le protocole de transfert de fichiers peut lui aussi varier, il existe le Secure CoPy (SCP) ou encore le File Transfer Protocol (FTP).
- Enfin, le dernier point de variation dans l'accès au système d'exploitation des machines est le langage d'interpréteur de commandes (ou shell). Là encore, il existe de nombreux langages différents comme le Bourne Shell, le C-Shell, ou encore le shell de Windows. La réalisation des étapes du cycle de vie du déploiement dépend de ces différents paramètres.
-

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## II.PARAMETRE DU DEPLOIEMENT

- En outre, le contenu des étapes du cycle de vie, en termes d'actions élémentaires, varie également en fonction de la nature du logiciel. C'est là l'une des difficultés les plus grandes du déploiement logiciel à l'heure actuelle. Tout d'abord, il existe différentes technologies pour concevoir des logiciels. Ainsi, les opérations élémentaires de déploiement d'un logiciel peuvent dépendre du langage de programmation dans lequel le logiciel est réalisé.



# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## II.PARAMETRE DU DEPLOIEMENT

- C'est là l'une des difficultés les plus grandes du déploiement logiciel à l'heure actuelle. Tout d'abord, il existe différentes technologies pour concevoir des logiciels. Ainsi, les opérations élémentaires de déploiement d'un logiciel peuvent dépendre du langage de programmation dans lequel le logiciel est réalisé.
- En outre, pour chacun de ces paradigmes, il existe une multitude d'implémentations.
- À titre d'exemple, on peut citer les modèles de composants Entreprise Java Beans (EJB), Le modèle EJB est fortement orienté vers la manipulation de bases de données,

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## II.PARAMETRE DU DEPLOIEMENT

- Chacune de ces technologies possède ses propres concepts et son propre processus de déploiement.
- De plus, les logiciels possèdent également une granularité différente.
- En effet, différentes entités aux rôles différents interviennent dans le déploiement de logiciels. Certains logiciels sont des bibliothèques de fonctions utilisées par les applications. D'autres sont des serveurs d'applications qui hébergent des applications.
- D'autres sont des serveurs d'applications qui hébergent des applications. Enfin, les applications métiers elles-mêmes sont des entités.

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## II.PARAMETRE DU DEPLOIEMENT

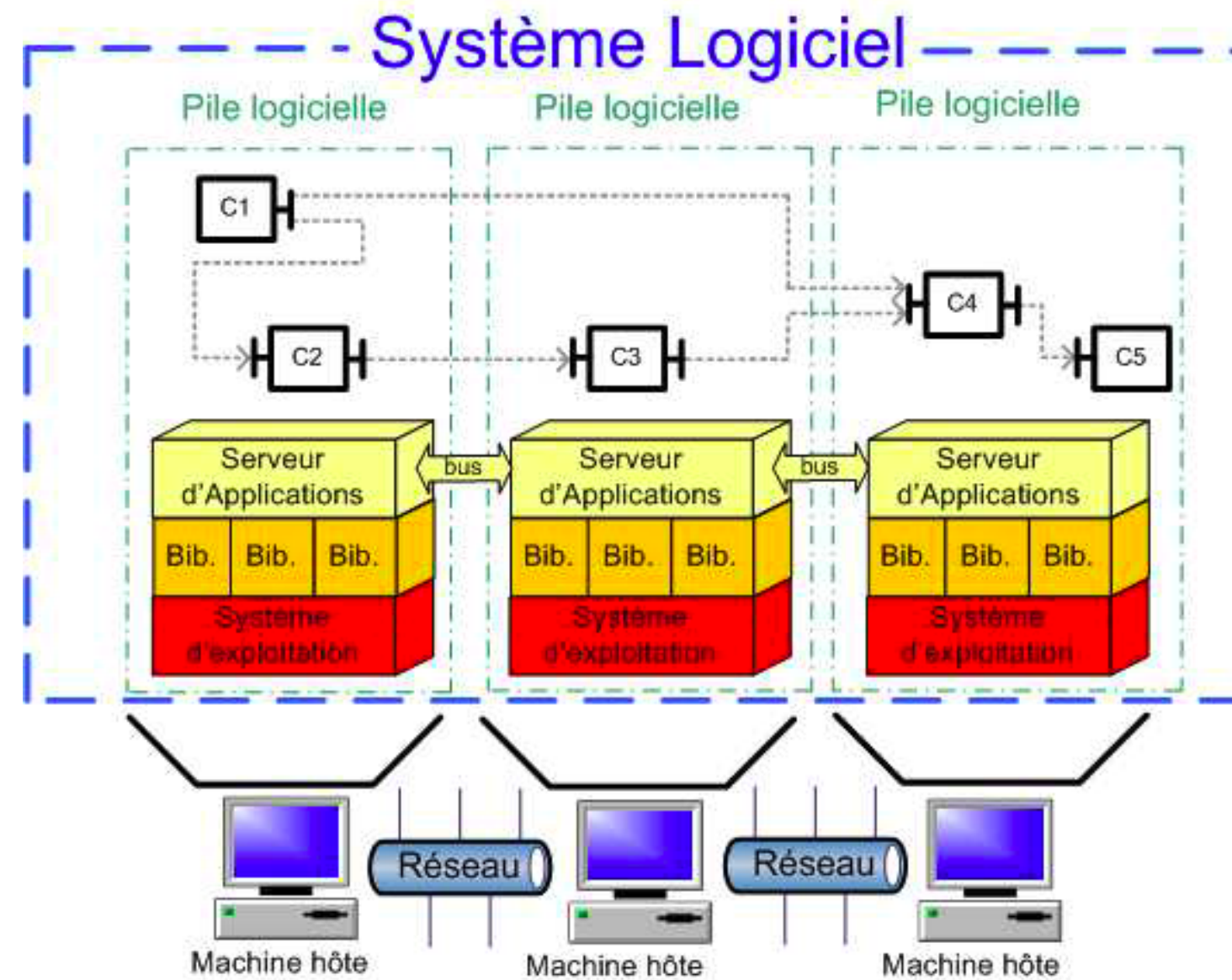
L'ensemble de ces entités de granularité différente constituent ce que nous appelons un système logiciel, et l'ensemble des logiciels d'un système déployés sur une machine est appelé la pile logicielle . Cette notion de système logiciel est représenté

Ainsi selon la granularité du logiciel que l'on déploie, les actions élémentaires de déploiement sont différentes.

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## II.PARAMETRE DU DEPLOIEMENT

. Cette notion de système logiciel est représentée sur la figure suivante :



Deploiement et mise en service/M.SENE

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## III. DESCRIPTION DU DEPLOIEMENT

- Les étapes du cycle de vie du déploiement varient selon les conditions matérielles—i.e. les paramètres de la machine hôte— ainsi que selon les paramètres du logiciel lui-même. Ainsi il faut posséder un moyen d'exprimer les différentes configurations et autres actions élémentaires à opérer. Par langage de déploiement nous entendons donc un langage dédié à la description du déploiement d'un système, qui permet d'exprimer une configuration de système logiciel.

-

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## III. DESCRIPTION DU DEPLOIEMENT

- Ces langages doivent permettre de décrire à haut niveau les propriétés d'un logiciel, les liaisons qu'il possède avec d'autres logiciels, ainsi que les actions à effectuer pour déployer ce logiciel sur une machine. De ces descriptions à haut niveau, il est possible de déduire les opérations à plus bas niveau à opérer concrètement sur les machines hôtes afin de rendre le système logiciel opérationnel.
- Cette abstraction doit permettre de décrire toutes les entités du système, quelle que soit leur granularité, mais aussi les machines hôtes ainsi que leurs caractéristiques. Il existe une multitude de langages discutés dans la suite du chapitre.

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## III. DESCRIPTION DU DEPLOIEMENT

- Ces langages doivent permettre de décrire à haut niveau les propriétés d'un logiciel, les liaisons qu'il possède avec d'autres logiciels, ainsi que les actions à effectuer pour déployer ce logiciel sur une machine. De ces descriptions à haut niveau, il est possible de déduire les opérations à plus bas niveau à opérer concrètement sur les machines hôtes afin de rendre le système logiciel opérationnel.
- Cette abstraction doit permettre de décrire toutes les entités du système, quelle que soit leur granularité, mais aussi les machines hôtes ainsi que leurs caractéristiques. Il existe une multitude de langages discutés dans la suite du chapitre.



# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## III. VERIFICATION DU DEPLOIEMENT

- Dès lors qu'un langage de déploiement est employé pour décrire le déploiement de systèmes

logiciels, la question de la vérification des descripteurs se pose. En effet, il faut pouvoir vérifier que la description abstraite du déploiement est correcte. En premier lieu, une vérification syntaxique est nécessaire afin de vérifier que les concepts utilisés dans le descripteur correspondent bien à des concepts définis dans le langage de déploiement et que leur emploi est valide.

- Ensuite, une vérification de typage peut être effectuée afin d'assurer que les types des concepts du langage ont été employés conformément à leur définition. Enfin, nous appelons le troisième type de vérification des vérifications sémantiques (ou comportementales) spécifiques au processus de déploiement logiciel.



# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## III. VERIFICATION DU DEPLOIEMENT

- Cette vérification doit survenir après les phases de vérification syntaxique et de typage. Elle consiste à assurer que le processus de déploiement décrit possède bien le comportement que l'on attend, sans effet de bord, ni erreurs survenant lors de l'exécution du processus de déploiement.
- Cette procédure vise à étudier, non plus la justesse de ce qui est écrit dans le descripteur, mais la cohérence des intentions exprimées dans ce descripteur.

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## III. VERIFICATION DU DEPLOIEMENT

À titre d'exemple, le travail d'un compilateur Java est de vérifier la syntaxe et l'usage des types dans un programme Java. Néanmoins, il existe un exemple de vérification sémantique dans les compilateurs Java : lorsqu'une instruction suit directement une instruction de type return, le compilateur relève une erreur. Le programme est syntaxiquement correct, il n'y a pas de problème de typage, mais il existe une incohérence dans l'intention du développeur, en l'occurrence une instruction qu'il a écrite ne pourra jamais être exécutée (à cause du return qui la précède directement).

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## III. VERIFICATION DU DEPLOIEMENT

Nous différencions trois types de vérifications différentes, dans le moment et la manière avec lesquelles elles sont notifiées.

- La vérification statique avertit l'utilisateur d'éventuelles erreurs avant de commencer la procédure de déploiement.
- La vérification dynamique directe prévient l'utilisateur à l'exécution d'une erreur précise rencontrée lors du déploiement.
- Enfin, la vérification dynamique indirecte prévient l'utilisateur d'une erreur rencontrée lors du déploiement, mais le message d'erreur notifié à l'utilisateur ne concerne pas directement le problème dans son déploiement, mais plutôt la conséquence de l'erreur faite par l'utilisateur (par exemple L'apparition d'une exception de type pointeur nul à cause d'un serveur non démarré).

# CHAPITRE 2 :CYCLE DE VIE DU DÉPLOIEMENT

## III. RECONFIGURATION DU DEPLOIEMENT

Le descripteur écrit dans un langage de déploiement est ensuite exécuté. Il en résulte un système entièrement déployé sur plusieurs machines hôtes. Néanmoins, nous l'avons vu dans le cycle de vie du déploiement, le système logiciel est sujet à d'éventuelles reconfigurations ou adaptations, avant son arrêt. Dès lors, le système doit connaître les informations qui lui permettront de savoir quand et comment il doit être adapté ou reconfiguré. En outre, le système logiciel doit avoir accès à des mécanismes de reconfiguration dynamique, c'est-à-dire capable d'analyser son environnement et d'exécuter des reconfigurations en fonction de cet environnement, le tout sans altérer son propre état.