

Ateliers de Génie Logiciel centrés procédés (AGLCP)

Mr K. Sellami

informatiqueii9@gmail.com



جامعة بجاية
Tasdawit n Bgayet
Université de Béjaïa

Environnement de de Génie logiciel

- Objectifs:

- Améliorer la fabrication du logiciel que ce soit en terme de



- Qualité,

- Délai,

- Productivité et

- Tous les paramètres entrant en jeu dans la chaîne de production des logiciels, ce qui conduit l'apparition du concept environnement de génie logiciel.



Apparition du concept environnement de génie logiciel.

Environnement de de Génie logiciel

Définition :

- ensemble intégré d'outils et de mécanismes permettant de supporter toutes les phases de développement du logiciel (analyse, conception, écriture de code, test,...etc.). (Sommerville 2000)

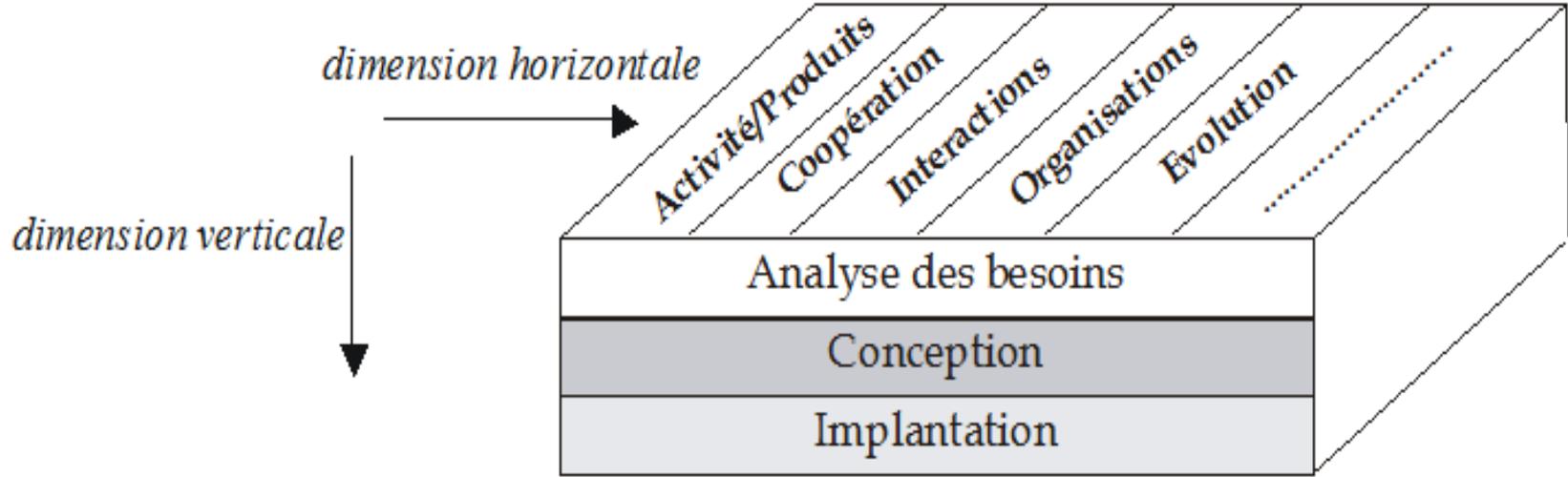
Travaux de recherche

- les modèles de cycle de vie du logiciel,
- les environnements intégrés de génie logiciel, et
- les ateliers (environnements) centrés procédés logiciel

Les procédés logiciels

Définition :

- la manière dont le développement de logiciel est organisé,



implémentation) ce qui rend le procédé logiciel difficile à gérer [Amiour 1999].

Les procédés logiciels

Types:

a. Les modèle de cycle de vie

- Ces modèles ont permis de mieux comprendre le procédé logiciel par la description des activités abstraites et de leurs ordonnancement, (Royce 1970) le premier à proposer un modèle de cycle de vie, depuis de nombreuses améliorations et modifications y sont apportées [Boehm 86].

Insuffisances

ces modèles ne tiennent pas compte

- de l'activation,
- de l'échec ou au contraire du succès des activités, et en général,
- de la synchronisation des activités entre elles,
- pas plus qu'ils ne s'intéressent aux artefacts manipulés par ces activités (Les ressources, l'organisation, la planification,...etc)

Les procédés logiciels

Types:

b. Les environnements intégrés de génie logiciel (AGL)

- Ces environnements se sont succédé aux modèles de cycle de vie pour offrir un support aux développements de logiciels,

Travaux réalisés: les ateliers de Génie logiciels (AGL) ; s'agissait d'environnements intégrés de génie logiciel qui sont un ensemble de fonctionnalités nouvelles reposant sur l'intégration d'outils souvent existant et connus des développeurs (compilateurs, éditeurs de liens, gestionnaires de fichiers, ...etc.), cette intégration permettait d'utiliser cette collection d'outils pour produire des logiciels avec plus de facilité.

- Le terme *développeur* dénote l'ensemble des personnes impliquées dans le développement d'une application : analystes, concepteurs, programmeurs, testeurs, etc.



Insuffisances

ces modèles restent limités malgré leurs succès relatif

- la définition, l'enchaînement des activités ne sont pas proposés (la façon dont le produit logiciel est fabriqué n'est pas explicité).

Les procédés logiciels

Types:

c. Les ateliers de génie logiciel centrés procédé (AGLCP)

- Nouvelle ère de recherche sur les procédés logiciels
- Un AGLCP est un système dans lequel la manière dont le logiciel est fabriqué (ou doit être fabriqué) est définie de façon explicite et avec suffisamment de détails.
- Ces environnements constituent la voie privilégiée de la recherche sur les procédés du génie logiciel ces dernières décennies, de nombreux travaux se sont focalisés sur les AGLCP : Marvel, Oz, Peace, Peace+ , SPADE Tempo, APPE, etc.

Les ateliers de génie logiciel centrés procédé (AGLCP)

Concepts

La terminologie associée au procédé logiciel concerne :

- **L'activité** : C'est une opération atomique ou une étape du procédé, par exemple : la conception ou la compilation.
- **Produit** : est une donnée manipulée ou résultante d'une activité, ça peut être composés d'autres produits, par exemple : code source, plan de teste.
- **Ressource** : Ce sont les moyens exigés par l'activité pour être exécutée, il y a deux types de ressource :
 - **Le développeur** : (agent humain) généralement directement lié à l'activité par son rôle.
 - **Le rôle** : Peut être utilisé pour définir les responsabilités et obligations du développeur
 - **La direction** : C'est le pilote « guide » qui peut être de règles ou des procédures qui gouverne l'activité.
 - **Les outils** : C'est l'équipement matériel ou logiciel dont l'activité a besoin pour son exécution comme par exemple : (éditeurs, compilateur, etc).

Les ateliers de génie logiciel centrés procédé (AGLCP)

Le modèle de procédé

Un modèle de procédé est une abstraction du procédé logiciel décrit de manière formelle ou semi formelle.

Dans un modèle de procédé on trouve [Bandinelli et al 1994] :

- La description des **ressources** (outils, acteurs, etc.) que requiert le procédé.
- Les **activités** et les **tâches** pour lesquelles le procédé est défini et structuré.
- **L'enchaînement** (ou ordonnancement) de ces **activités** ou **tâches**.
- Les **informations nécessaires** à la définition du procédé.

Les ateliers de génie logiciel centrés procédé (AGLCP)

Différents PML se sont utilisés, nous pouvons distinguer 6 paradigmes importants

- **L'approche procédurale**, elle décrit le procédé sous forme de programme. Pour cela, elle utilise des langages de programmation classiques étendus par des concepts relatifs aux procédés. (ex : systèmes ARCADIA [Osterweil 1987], ProcessWise, etc.)
- **L'approche déclarative** ; dans cette approche, le procédé est décrit par des règles Pré-condition/Action/Post-condition. La pré-condition est une contrainte nécessaire à l'exécution de l'action tandis que la post-condition représente l'effet de la condition. (Marvel [Kaiser et al. 1988], Merlin, etc.)
- **L'approche orientée but adoptée par l'environnement** Peace [Arbaoui et al. 2002] et Peace+ [Alloui et al. 1994], SCALE [Oquendo 1995].
- **L'approche basée sur les réseaux de Pétri** illustré par les systèmes Process Weaver, SPADE [Bandinelli et al. 1992] et Leu, [Sellami et al. 2011a].
- **L'approche fonctionnelle** ; le procédé est décrit par une hiérarchie de fonctions **mathématiques** qui représente des relations entre des données d'entrées et des données de sortie, il est illustré par les systèmes HFSP et à PDL, etc.
- **L'approche mutli-paradigme** ; au lieu d'utiliser un formalisme unique pour la **description** de tous les éléments du procédé, cette approche combine deux ou plusieurs paradigmes, elle est prise en compte dans les systèmes comme Alf, APEL [Amiour 1999, Estublier et al. 1998], etc.

Les ateliers de génie logiciel centrés procédé (AGLCP)

Formalisation du procédé

Dans le cycle de vie d'un procédé logiciel, on peut distinguer trois phases :

- La phase de spécification des besoins
- La phase de conception, de modélisation
- La phase d'implantation du procédé tel qu'il a été défini.

Pour couvrir ce cycle de vie, trois catégories de langages ont été proposées :

- Un langage de spécification
- Un langage de modélisation
- Un langage d'implantation

Les ateliers de génie logiciel centrés procédé (AGLCP)

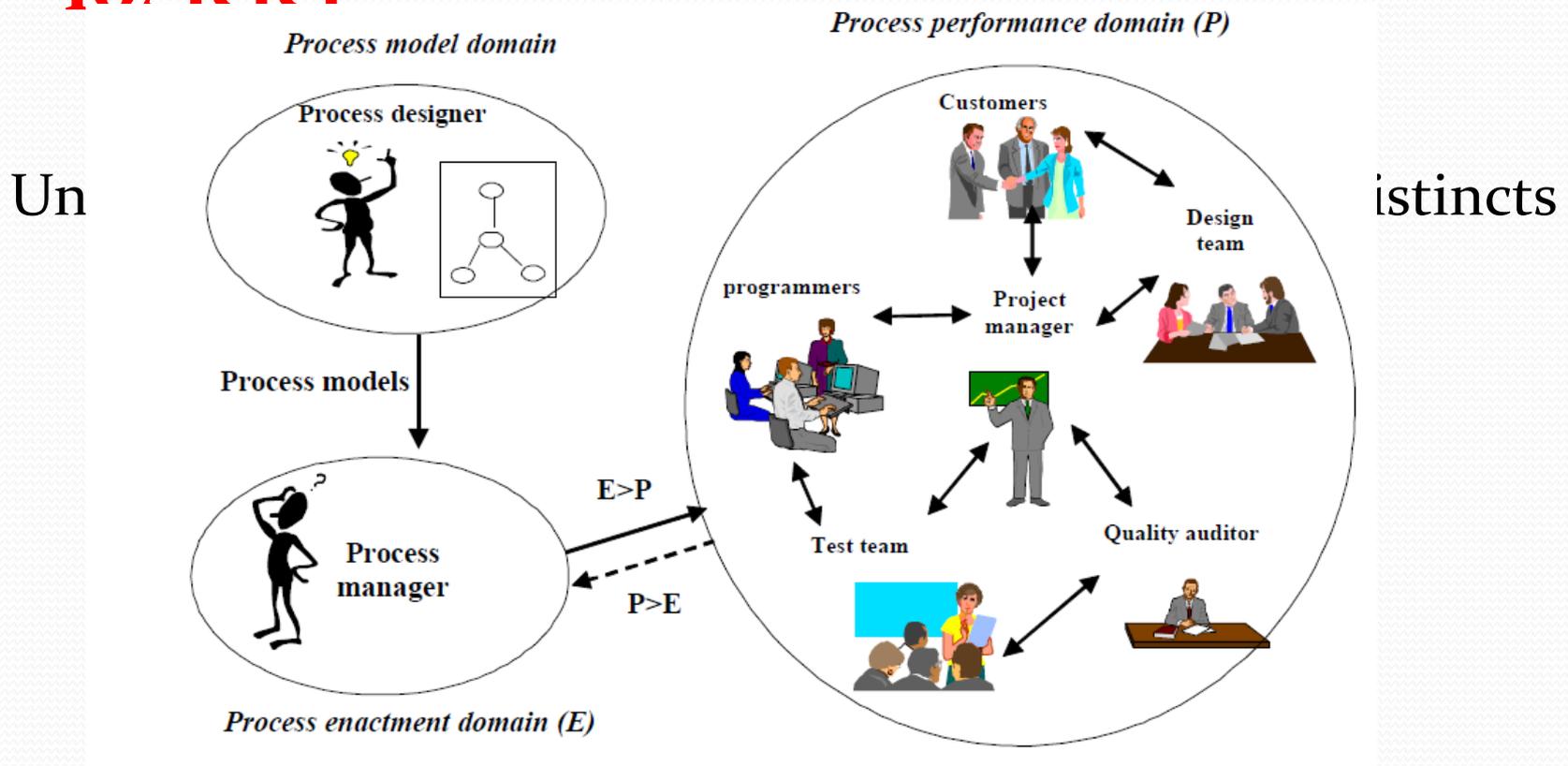
Formalisation du procédé (règles)

Un formalisme d'un procédé doit être assimilable par les différents acteurs (manager, développeurs, etc.), c'est pour cela qu'un formalisme doit être de haut niveau et doit satisfaire les besoins suivants :

- Il doit être exécutable.
- Il doit permettre de décrire et de supporter l'ensemble du cycle de vie du processus ainsi que tous ses niveaux d'abstraction,
- Il doit prendre en compte la description dynamique de l'ordonnancement des activités du procédé,
- Il doit permettre de supporter et de modéliser l'évolution des procédés et de leurs modèles,
- Il doit permettre de modéliser et de gérer l'incertitude et l'incomplétude de certaines informations,
- Il doit prendre en charge le support de la communication, la coordination, la négociation et la coopération entre les divers participants dans le procédé.

Les ateliers de génie logiciel centrés procédé (AGLCP)

Les différents domaines du procédé logiciel



Les ateliers de génie logiciel centrés procédé (AGLCP)

Les aspects d'un environnement d'ingénierie logiciels efficace :

l'arrivée de l'Internet et l'évolution des traditions de travail des développeurs de logiciels ; les environnements d'ingénierie logiciels doivent répondre à des exigences grandissantes des développeurs et évoluer pour mieux exploiter les outils et nouvelles technologies qui s'offrent à eux.

Les ateliers d'ingénierie logiciels centrés procédés logiciels actuels sont caractérisés par ;

1. L'AGLCP peut être distribué géographiquement tel que chaque groupe de travail peut être dans *un* lieu distant (une autre ville ou un autre pays) ; il peut développer un composant à n'importe quel moment de la journée sans se soucier des autres groupes de travail.
2. Les modèles de procédés peuvent être hétérogènes un environnement d'ingénierie logiciel doit être capable de supporter et d'exécuter tous types de procédés logiciels.
3. L'interaction, la communication, la coordination, prennent une place très importante et doivent être pris en charge de manière efficace car la plupart du temps de développement est consommé dans l'échange d'informations entre les développeurs.

Les ateliers de génie logiciel centrés procédé (AGLCP)

Les aspects d'un environnement d'ingénierie logiciels efficace :

6. L'AGLCP doit gérer l'évolution dans des modèles de procédés, et doit supporter leur changement dynamique pendant l'exécution.
7. AGLCP doit pouvoir utiliser tous types d'outils ou de langages, les intégrer facilement sans remettre en cause tout l'environnement et son fonctionnement.
8. Dans le cas où l'AGLCP est distribué, les espaces de travail distribués doivent avoir une liberté locale et pouvoir agir sur les tâches locales sans intervention hiérarchique ; d'autre part un certain contrôle global pour sauvegarder la consistance et la cohérence des modèles de procédés en exécution est exigé.
9. L'AGLCP doit avoir le moyen de prendre en charge toutes déviations des modèles de procédés exécutés par rapport aux modèles de procédés instanciés de cela de manière dynamique.
10. La plus part des informations et connaissances qui circulent dans l'AGLCP doivent être partagées ; elles sont utilisées par plusieurs développeurs, il est possible de les modifier, les supprimer ou les déplacer selon le niveau d'accès de l'utilisateur.

La voie des AGLCP fédéré

1- La réutilisation de composants dans les AGLCP

- La plupart des travaux liés au domaine des AGLCP s'intéresse à un aspect particulier entre autres : la distribution de l'environnement, la coopération, la coordination des composants, etc. d'autres ont défini des AGLCP comme des fédérations de composants qui doivent interopérer.

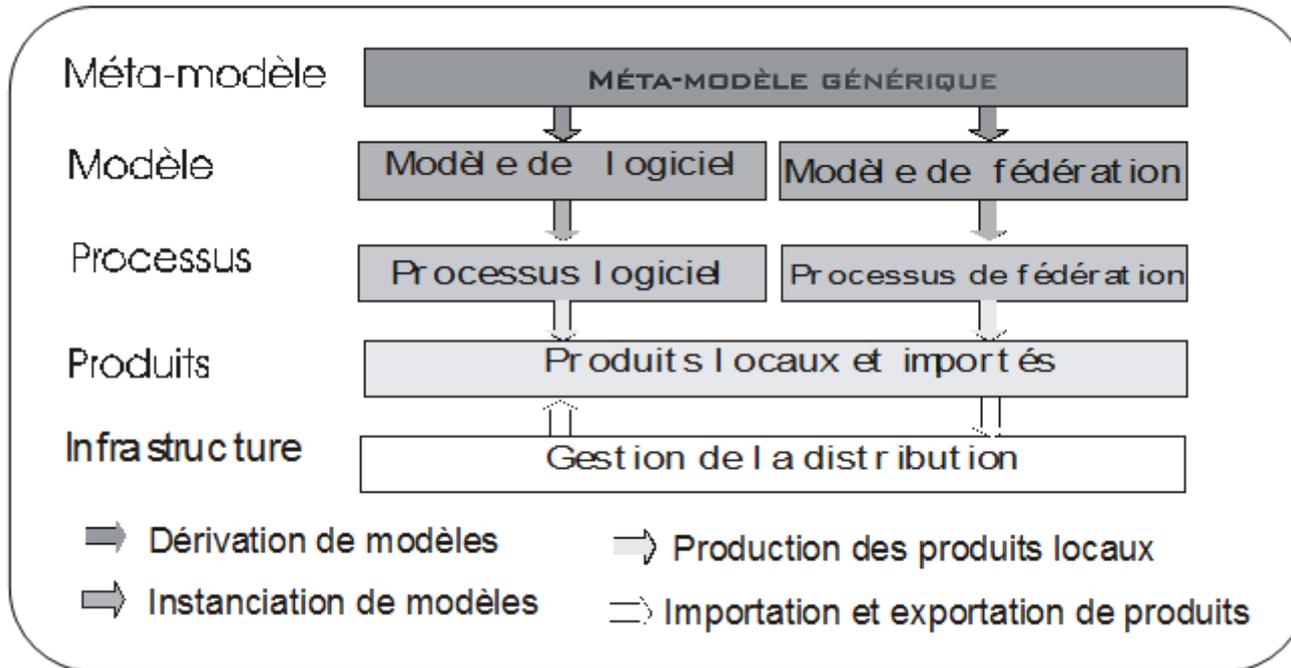
Exemple: Endeavour :

- Le système Endeavours [Bolcer et Taylor 1996] est un système distribué dont le but est d'assister les procédés logiciels et les procédés de workflow. Le principe général est l'utilisation des composants appelés ; "handlers", écrits en langage Java, qui sont l'implémentation des modèles. Au niveau exécution l'approche est hétérogène (utilisation d'outils différents).

La voie des AGLCP fédéré

2- Vers des

- La fédér
- La mod
- En fait,



posants de
composant
rme des
.CP de
autre

- Exemple

travaux de l'équipe de M. Jacky Estublier de l'université de Grenoble) , PIE (*Process Instance Evolution*)

uit des

Bilan et conclusion

- L'objectif principal des AGLCP était l'assistance des procédés logiciels et cela en proposant une vision « unifié » des outils utilisés.
- Cependant, les environnements qui sont ouverts et flexibles et qui permettent l'intégration d'outils de type COTS dans l'environnement tout en restant disponible par ailleurs (préservation de l'autonomie), sont rares.
- Le tableau suivant donne une comparaison de quelques AGLCP vu précédemment en tenant comme critère de différenciation leurs hétérogénéité, la distribution, l'implication d'un modèle commun (Fédération), la possibilité d'intégration de composants commercial de type COTS, ainsi que l'évolution des constituants. Abréviations anglo-saxonnes de « commercial off the shelf », ce sont des composants logiciels disponibles sur le marché.

AGLCPs	Facettes techniques				
	Interopérabilité	Hétérogénéité	Fédération	Intégration de COTS	Evolution
Provence	Non	Oui	Non	Oui	Non
Endeavors	Non	Oui	Non	Non	Non
Peace/Peace+	Oui	Non	Non	Non	Oui
APPEL	Oui	Non	Oui	Non	Oui
PIE	Oui	Oui	Oui	Non	Oui
OZ	Non	Non	Oui	Non	Oui

Bilan et conclusion

Les apports du domaine des AGLCP :

- Diriger le développement de logiciel grâce au procédé logiciel
- Structurer le processus de développement par un formalisme permettant l'identification des concepts, et la définition des modèles
- La mise en synergie de fonctionnalités hétérogènes pour satisfaire les besoins des développeurs.

Les limitations :

- Complexité, redondance et insuffisance des supports : les AGLCP sont des environnements de plus en plus complexes car ils intègrent de nombreuses fonctionnalités.
- Rigidité, monolithisme et faible évolution des supports : les AGLCP sont des environnements qui sont générés à partir d'une description. L'évolution de tels supports impose l'évolution de la description. Toute modification de la description induit une re-génération de l'environnement qui ne peut se faire en cours d'exécution ;
- Faible hétérogénéité des constituants : les AGLCP "modulaires" intègrent rarement des composants hétérogènes ;
- Faible autonomie des constituants : l'intégration de composants est forte et ne permet pas à ces derniers de préserver leur autonomie et de pouvoir être utilisés par ailleurs ;
- Peu (pas) d'AGLCP qui intègrent des composants quelconques issus du marché (COTS) : les outils intégrés par les AGLCP sont bien souvent de type "load and go".

Conclusion

L'objectif à venir des recherches dans le domaine des environnements de génie logiciel sera alors de proposer un environnement flexible, adaptable, ouvert. Les approches présentées considèrent l'outil comme simple ressource ou alors comme acteur d'une activité : il n'est pas pris en compte comme un concept majeur, sa dimension est donc réduite, alors que la participation des outils logiciels ne cesse de croître, et les problèmes qui se posent au niveau des outils comme le recouvrement de concepts entre les outils, la cohérence des modèles opérationnels des différents outils, la gestion de la synchronisation, etc. sont totalement occultés.