Composants logiciels

Conception orientée par les modèles

Réutilisation, certification

[Introduction](#)

[Source](#)

[Relevé de quelques éléments](#)

Introduction ^(^)

Ce dossier traite de la démarche informatique consistant à développer les logiciels sous forme de « **briques logicielles réutilisables** » afin ensuite de pouvoir « intégrer ces briques dans des développements spécifiques ».

Source ^(^)

Ce dossier repose sur l'ouvrage « **Technologies clés 2010** » du **Ministère de l'Economie des Finances et de l'Industrie** (<http://www.industrie.gouv.fr>) et plus précisément dans cet ouvrage sur l'exemple de technologies clés 2010 : « Composants logiciels ».

Présentation de l'ouvrage : « Comme les deux éditions précédentes, l'ouvrage Technologies clés 2010 se propose de dresser un inventaire complet des techniques de pointe qui devraient permettre à l'économie française de rester compétitive au cours des années à venir » [Extrait le 27/10/06 de l'URL <http://www.industrie.gouv.fr/biblioth/docu/kiosque/cahiers/c116p1.html>].

Références de l'exemple « Composants logiciels »

Présentation de la troisième édition de l'étude prospective
« **Technologies clés 2010** »

Annexes - Exemples de Technologies clés 2010

Partie « **Composants logiciels** » (pages 35 à 37)

Bercy, 18 septembre 2006

Ministère de l'Economie des Finances et de l'Industrie

Accès URL :

<http://www.industrie.gouv.fr/infopres/presse/DP-Technocles-annexes.pdf>

Relevé de quelques éléments ^(^)

Il est repris ci-dessous – sans aucune complétude - quelques éléments de [la partie](#) « Composants logiciels », dont il est fait des citations (texte entre guillemets), à laquelle se reporter directement pour plus de détails.

Techniques de génie logiciel

MDA (Model Driven Architecture), conception orientée par les modèles : « A l’instar de l’initiative MDA (Model Driven Architecture), l’ingénierie des modèles propose de capturer la logique d’une application au travers d’une modélisation ce qui facilite les procédures de tests et permet la génération du code. Chaque spécialiste apporte sa plus-value métier et un architecte logiciel met en musique l’ensemble des briques logicielles métiers ». L’ensemble est accompagné d’ « outils permettant la modélisation, la vérification de l’application, la génération du code et la réutilisation optimale des briques logicielles ». « Sur la base de spécifications en langage formel adapté au domaine d’application, un modèle indépendant de la plate-forme d’exécution est réalisé. Ce modèle peut alors être transformé en un ou plusieurs modèles spécifiques à la plate-forme, testé, ou encore utilisé pour générer automatiquement du code ».

Programmation Orientée Aspect (AOP) : « La Programmation Orientée Aspect (AOP) est une technique d’ingénierie logicielle » dont le but « est de faciliter la conception [des] logiciels ‘par morceaux’ ».

Composants logiciels

Notion de « composants logiciels » : « Un composant est une brique logicielle réutilisable par plusieurs applications. Conceptuellement, un composant correspond à l’expression logicielle des caractéristiques et des comportements d’un objet, d’un service, d’un aspect ou éventuellement d’une application informatique complète. Les composants logiciels tendent à devenir des agents autonomes, capables d’apprendre, de s’organiser, de découvrir les services offerts par les autres composants ».

Mise en pratique : agencement/échanges entre composants logiciels

« Le problème de l’**intégration** n’est pas évident : il faut notamment s’assurer que le comportement du composant sera conforme à ce que l’application attend. Pour cela, il est nécessaire de développer des méthodes et des outils pour assister le développement et l’intégration de composants : capture des caractéristiques sémantiques, accès à des bases de données de composants documentés, test de conformité et d’intégrité du composant ».

Normalisation, certification

« Des efforts importants de **normalisation** au niveau mondial ont été réalisés au cours de ces dernières années, particulièrement sur la base d’**UML** pour la modélisation et de **XML** pour la description, rendant possibles les applications actuelles. »

« La seule normalisation n’est cependant pas suffisante. Elle doit s’accompagner de politiques de **transparence** sur certaines parties de code devant servir d’interface (*Open Source*). En particulier, l’indépendance des applications vis-à-vis des composants logiciels, et des éditeurs de ces derniers, est un enjeu majeur pour la pérennité des développements et peut nécessiter, dans certaines circonstances, le recours à des **tiers de confiance**. »

« Il sera nécessaire [...] qu'un cadre pour la **certification** des briques logicielles réutilisables soit établi ». « La certification des composants logiciels est un enjeu central dans le développement de l'économie autour du logiciel libre ».

- *mise en ligne le 02/11/06* – [La page au format pdf](#) (02/11/06)

Plate-forme INRA-ACTA-ICTA, Modelia <http://www.modelia.org>