

Les types de programmeurs :

Système : développement des langages de programmation

Application : utilisent ces langages pour créer de la valeur ajoutée à des fins commerciales.

Les programmeurs d'applications se

concentrent sur leurs projets sans se soucier des techniques et mécaniques de bas niveaux.

Les programmeurs

d'applications utilisent des bibliothèques ou outils appelés : framework.

Un **framework** est un ensemble de bibliothèques, d'outils et de règles à suivre, qui aident au développement d'applications.

Les **frameworks** sont développés par des programmeurs système.

Un framework est composé de plusieurs briques/composants qui sont en interaction les uns avec les autres. Les applications peuvent être écrites de manière plus efficace si nous utilisons un framework adapté au projet au lieu d'être obligé de réinventer à chaque fois la roue. Un framework Java fournit un ensemble de fonctionnalités à partir d'une implémentation objet. Lors de développement à grande échelle et de conception par équipe, les frameworks sont alors très utiles, voire indispensables.

types de frameworks :

- les frameworks d'infrastructure système, qui permettent de développer des systèmes d'exploitation, des outils graphiques et des plateformes web (Struts, Spring...) ;
- les frameworks communicants (appelés intergiciels) ;
- les frameworks d'entreprise (développements spécifiques) ;
- les frameworks de gestion de contenu (type Content Management System).

Pourquoi les frameworks :

- Standardisation
- combler les manques ou faiblesses de l'API Servlet et JavaServer Pages (JSP)

les développeurs utilisent les Servlets pour les traitements, et les pages JSP pour l'affichage.

pour une architecture multiniveaux de type MVC, les Servlets représentent l'aspect Contrôle, les Modèles l'accès aux données et les JSP la partie Vue. Ce modèle est plus complexe à développer mais beaucoup plus simple à tester, à maintenir et faire évoluer. De même, la notion de réutilisabilité est mise en avant avec ce type de modèle.

STRUTS 2 repose sur le modèle MVC 2 :

- un système évolué de gestion du routage ou navigation ;
- un système de validation de formulaires et d'entrées, simple à mettre en œuvre ;
- un système puissant de plugins ou d'extensions (pour les graphiques, sources de données...) ;

- la gestion de l'internationalisation pour le développement de sites multilingues ;
- le support de la technologie Ajax ;
- un outil de débogage en standard ;
- une bibliothèque puissante de balises.

Struts 2 propose (tout comme Struts 1 d'ailleurs) d'utiliser un minimum de **règles de conceptions** qui ne sont malgré tout pas obligatoires :

- Ne pas utiliser de code 100 % Java dans les pages JSP. Toute la logique de contrôle étant placée dans les classes d'actions (Servlets).
- Utiliser des bibliothèques de balises pour accéder aux objets et parcourir des collections.
- Écrire le minimum de code répétitif et utiliser les outils proposés par le framework