

INF5153 – Génie logiciel : conception

Conception architecturale : UML

Jacques Berger

# Objectifs

Introduire les diagrammes de conception architecturale

# Prérequis

UML

# Diagramme de déploiement

Le diagramme de déploiement illustre la disposition physique des composants

Un noeud représente une machine, ou un appareil pouvant exécuter un logiciel

Un noeud peut contenir des composants (des logiciels) et des artéfacts (fichiers, bases de données, etc.)

# Diagramme de déploiement

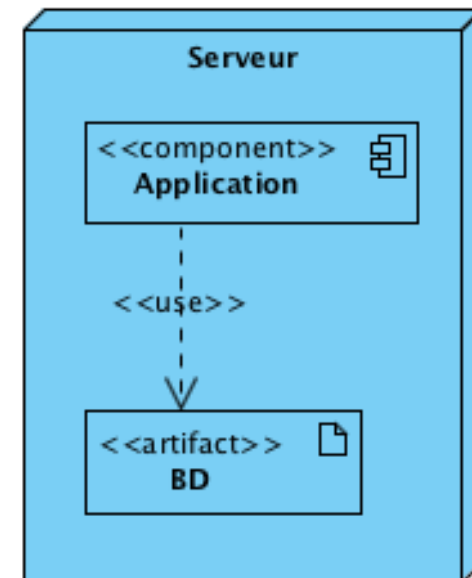
Un noeud est représenté par une boîte avec le nom du noeud



# Diagramme de déploiement

Si le noeud exécute un logiciel, le logiciel est indiqué comme un composant

Différentes relations peuvent exister entre les noeuds, les composants et les artéfacts



# Diagramme de déploiement

Voir un exemple

# Diagramme de packages

Illustre les packages qui composeront le logiciel et les dépendances entre eux

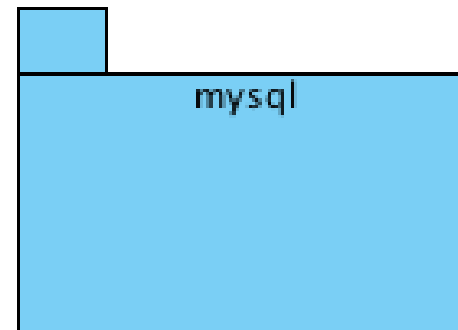
Les fonctionnalités d'un package doivent avoir un but commun

Une bonne décomposition en packages améliorera la maintenabilité d'un logiciel



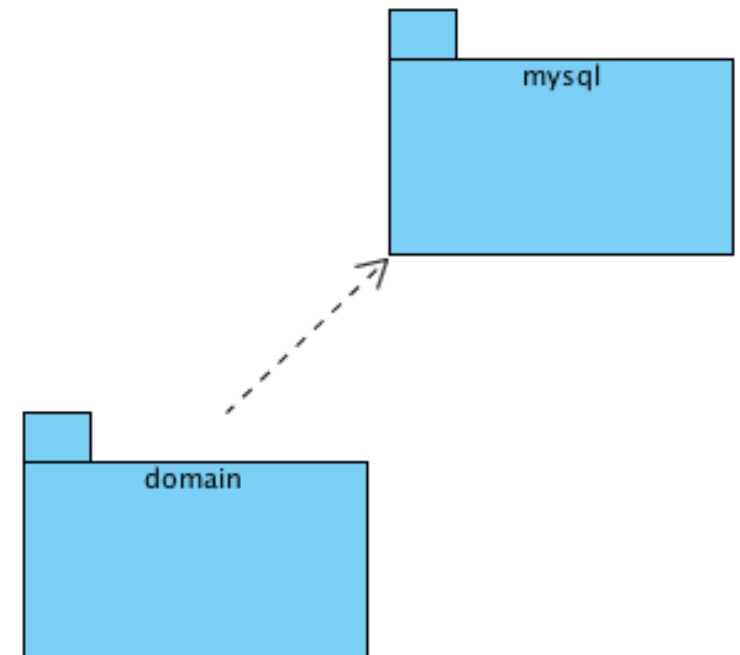
# Diagramme de packages

Un package est représenté par une boîte avec un onglet



# Diagramme de packages

Une dépendance entre deux packages est représentée par une flèche pointillée



# Diagramme de packages

Idéalement, il faudrait éviter les dépendances inverses et les dépendances circulaires

# Diagramme de packages

Voir un exemple

# Diagramme de composants

Un composant est un logiciel ou un module d'un logiciel

Les composants communiquent entre eux au travers d'interfaces

Un composant peut offrir une interface, ou en utiliser une

# Diagramme de composants

Voir un exemple

# Plus loin...

The Unified Modeling Language Reference Manual, Second Edition  
James Rumbaugh, Ivar Jacobson, Grady Booch  
Addison-Wesley, 2005