



CENTRALE L Y O N	<h2>Conception d'Architecture</h2>
Bertrand DAVID : UML et le Développement de Logiciels	<p>Objectif: Définir un découpage structurel de l'application pour permettre le travail coordonné et parallèle.</p> <ul style="list-style-type: none"><li>→ découper l'application pour répartir le travail de conception</li><li>→ appliquer une méthode de conception pour définir les modules</li><li>→ rédiger le dossier de conception d'architecture</li><li>→ compléter le plan de tests de validation</li></ul>
BTD/UML-DL	3

CENTRALE L Y O N	<h2>UML en phase de conception</h2>
Bertrand DAVID : UML et le Développement de Logiciels	<p>Objectif : passer du <i>quoi</i> au <i>comment</i></p> <p>Deux niveaux :</p> <ul style="list-style-type: none"><li>● La conception du système (conception d'architecture)</li><li>● La conception des objets (conception détaillée)</li></ul> <p>Deux approches :</p> <ul style="list-style-type: none"><li>● Architecture prédéfinie</li><li>● Architecture à concevoir</li></ul>
BTD/UML-DL	4

CENTRALE  
L Y O N

Bertrand DAVID :  
UML et le Développement de Logiciels

## La conception du système :

- Décomposition en sous-systèmes
- Détermination des priorités
- Gestion des données

L'architecture prédéfinie d'un système est organisée en trois couches représentant trois sous-systèmes de " haut niveau " :

<b>Interface Homme-Machine</b>
<b>Objets métier (du domaine applicatif)</b>
<b>Infrastructure (base de données, gestion distribuée,...)</b>

BTD/UML-DL

5

CENTRALE  
L Y O N

Bertrand DAVID :  
UML et le Développement de Logiciels

## L'architecture à concevoir s'organise à l'aide de packages

- Détermination des priorités :
  - la robustesse
  - la sécurité
  - le type d'exploitation (monotâche, multitâche - préemptif ou non préemptif)
  - la portabilité
- Gestion des données :
  - approche par fichiers
  - utilisation d'un SGBD

BTD/UML-DL

6

CENTRALE  
L Y O N

Bertrand DAVID :  
UML et le Développement de Logiciels

## La conception des objets (pour préparer l'implémentation)

- Ajouter aux modèles objet de la spécification des détails liés à l'implémentation du système.
- Transposer le modèle objet de spécification par ajout ou par suppression d'entités en fonction du langage de programmation et du système de gestion de données utilisés.
- « Typer » des attributs : trouver les types primitifs (chaîne, entier,...) et des méthodes les manipulant

BTD/UML-DL 7

CENTRALE  
L Y O N

Bertrand DAVID :  
UML et le Développement de Logiciels

## Traduction des associations

- Lors de la conception il s'agit de remplacer des associations pour de nouveaux attributs dans les classes reliées
  - Exemple d'une association unidirectionnelle
  - Prise en compte des cardinalités :
    - 0..1, 1 simple pointeur
    - N ensemble de pointeurs
    - collection : tableau, liste chaînée, pile,...
  - Première solution : un nouvel attribut *œuvre*
  - Deuxième solution : un objet intermédiaire *dicoAuteurOeuvres*

BTD/UML-DL 8

CENTRALE  
L Y O N

Bertrand DAVID :  
UML et le Développement de Logiciels

BTD/UML-DL

## Traduction des agrégations

- Traduire qu'un objet agrégé peut être référencé par plusieurs objets agrégats,
- Détruire les objets agrégés quand l'agrégat est détruit, détruire l'objet agrégat quand les agrégés sont détruits
- Traduction des classes d'association souvent sous forme de dictionnaire.
- Transposition du modèle dynamique à l'aide d'un moteur dirigé par une table décrivant le graphe.

9

CENTRALE  
L Y O N

Bertrand DAVID :  
UML et le Développement de Logiciels

BTD/UML-DL

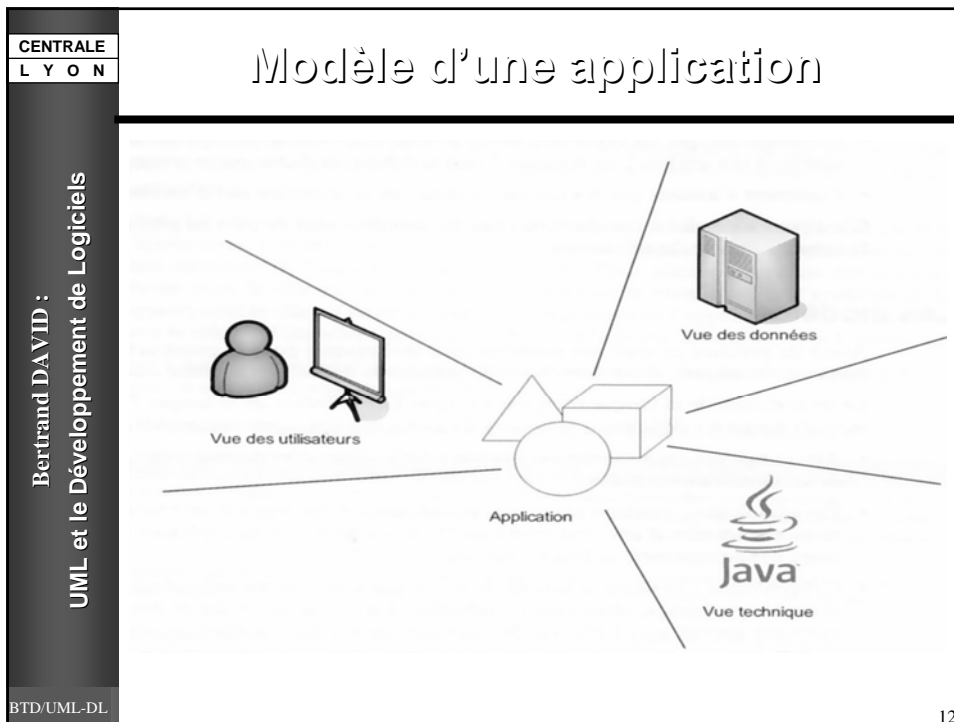
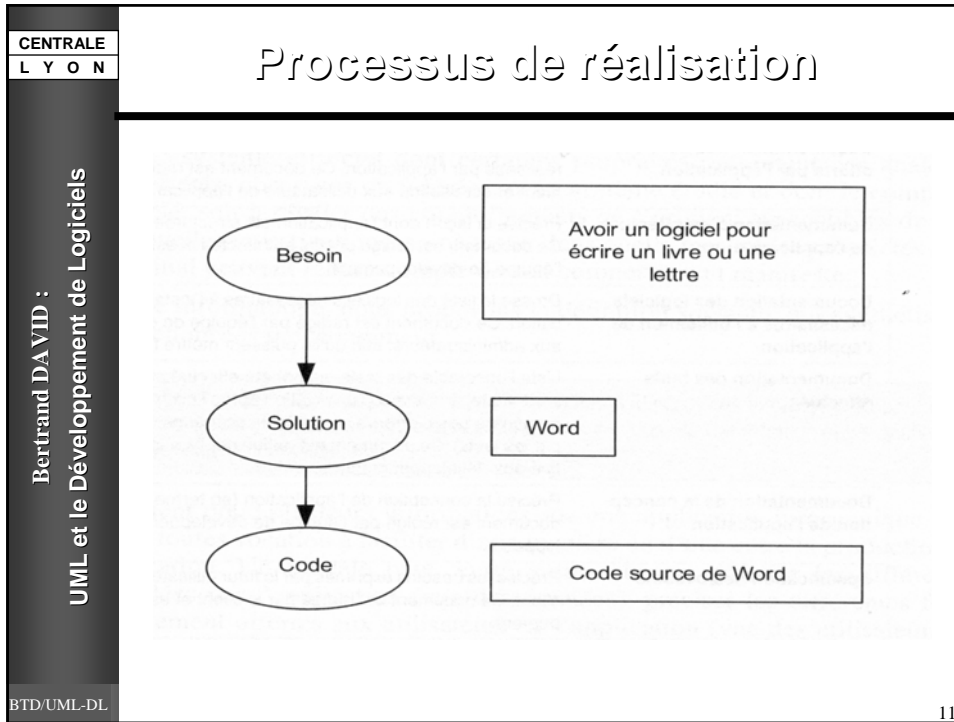
## Activités pour construire une application

```

graph TD
    Besoin --> Analyse
    Besoin --> Evolution
    Besoin --> Reutilisation
    Analyse --> Architecture
    Architecture --> Realisation
    Evolution --> Impact
    Impact --> Realisation
    Reutilisation --> Impact
    Reutilisation --> Separation[Separation du developpement]
    Separation --> Integration
    Integration --> MiseEnProduction[Mise en production]
    Realisation --> Tests
    Realisation --> Solution
    Solution --> MiseEnPreproduction[Mise en préproduction]
    Solution --> MiseEnProduction
    MiseEnProduction --> MiseEnPreproduction
    MiseEnPreproduction --> TestMontee[Test de montée en charge]
    
```

Figure 1.9

10



CENTRALE LYON

Bertrand DAVID : UML et le Développement de Logiciels

### Concepts de modèle UML

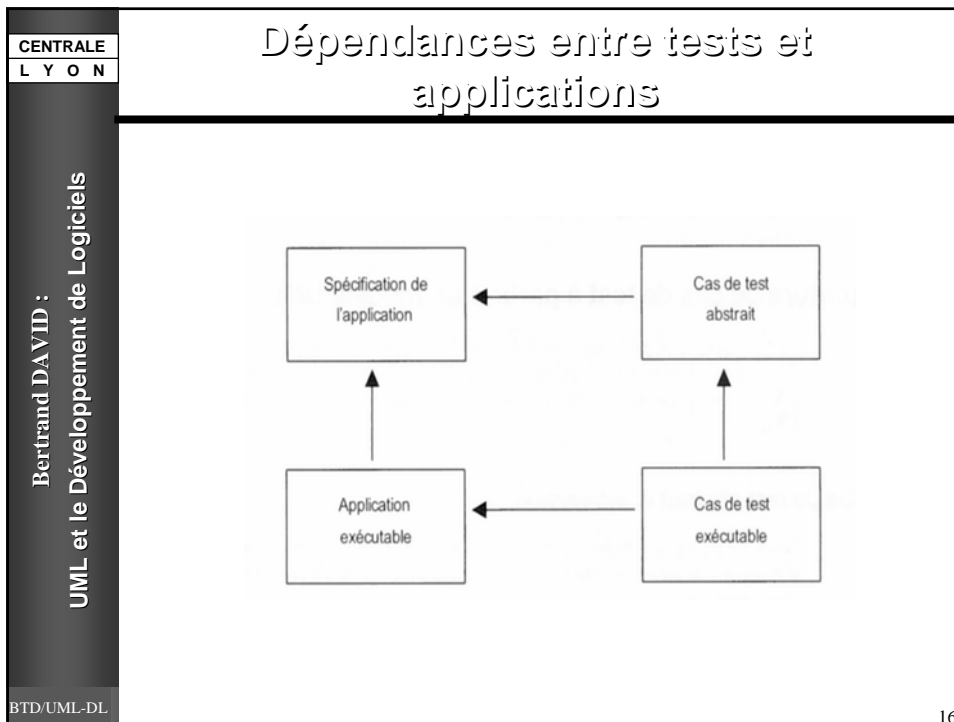
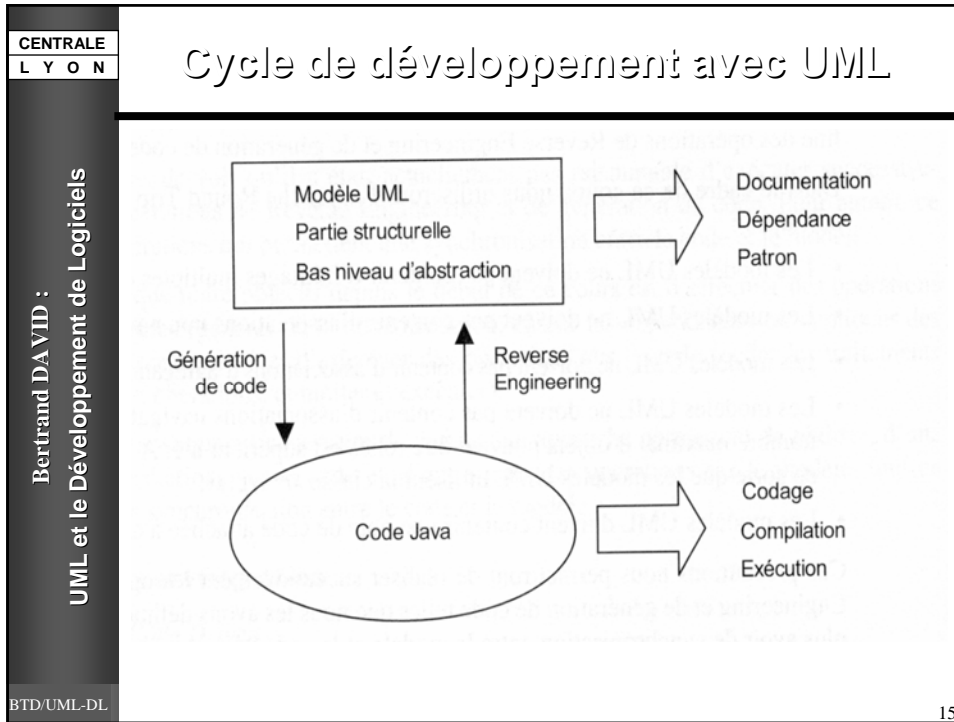
The diagram illustrates the relationship between UML views and code. At the top, a horizontal double-headed arrow labeled 'VUES' spans three boxes: 'Structure', 'Comportement', and 'Fonctionnalité'. Below these, a grid of boxes is connected by lines, with a vertical double-headed arrow on the left labeled 'ABSTRACTION' pointing upwards. The word 'COHERENCE' is written in the center of the grid. Below the grid, a lightning bolt icon points down to a rounded rectangle labeled 'CODE'. The footer contains 'BTD/UML-DL' and the number '13'.

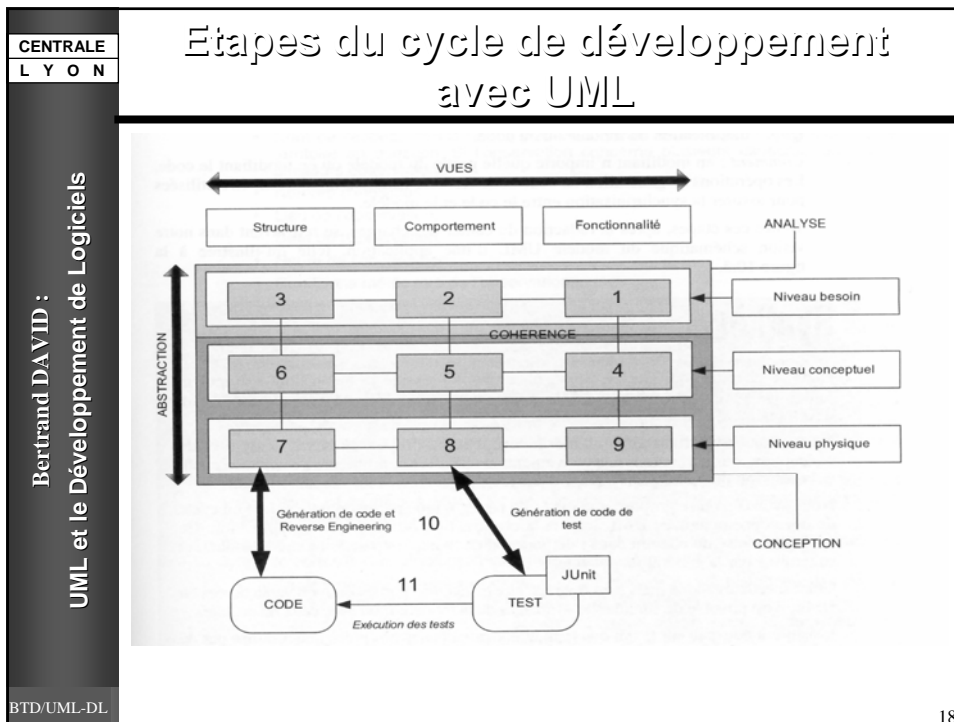
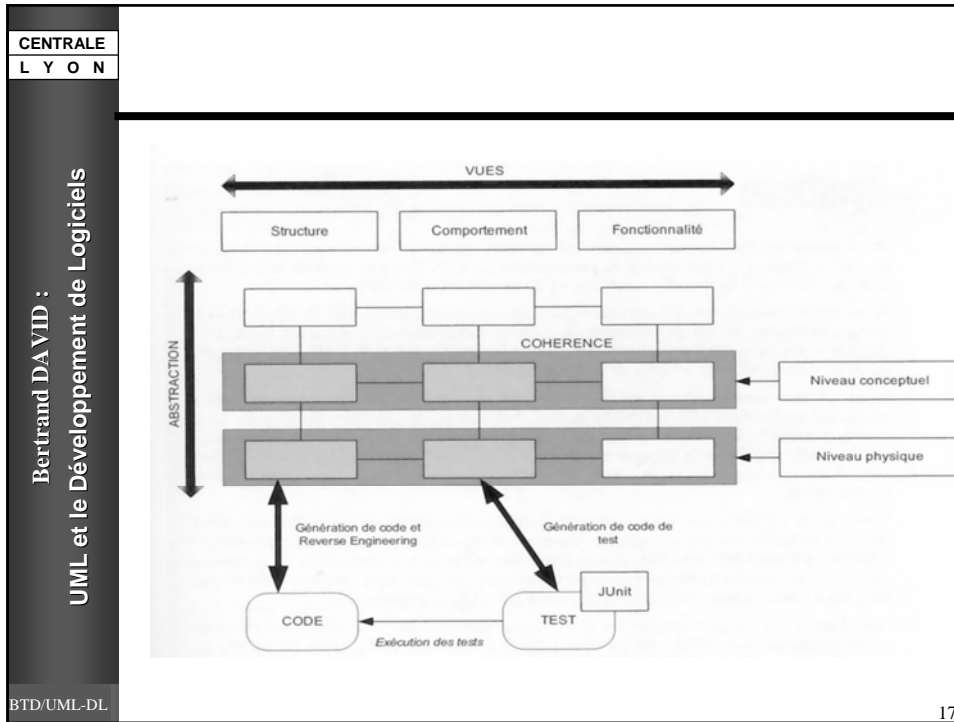
CENTRALE LYON

Bertrand DAVID : UML et le Développement de Logiciels

### Relation entre sémantiques UML et Java (modélisation – langage de programmation)

A Venn diagram with two overlapping circles. The left circle is labeled 'UML' and contains a box for 'Association'. The right circle is labeled 'Java' and contains boxes for 'API' and 'Traitement des opérations'. The overlapping area contains boxes for 'Classe', 'Objet', and 'Héritage'. The footer contains 'BTD/UML-DL' and the number '14'.





CENTRALE L Y O N	<h2>Étapes du cycle de développement avec UML</h2>
Bertrand DAVID : UML et le Développement de Logiciels	<ol style="list-style-type: none"><li>0. Rédaction du Cahier des charges</li><li>1. Analyse (niveau besoin) – cas d'utilisation</li><li>2. Analyse (niveau besoin) – séquences</li><li>3. Analyse (niveau besoin) – classes</li><li>4. Conception (niveau conceptuel) – cas d'utilisation</li><li>5. Conception (niveau conceptuel) – séquences</li><li>6. Conception (niveau conceptuel) – classes</li><li>7. Conception (niveau physique) – classes</li><li>8. Conception (niveau physique) – séquences</li><li>9. Conception (niveau physique) – cas d'utilisation</li><li>10. Génération de code et de tests</li><li>11. Compilation et exécution du code et des tests</li><li>12. Modifications de l'application (correction de bogues ou réalisation d'évolutions)</li></ol>
BTD/UML-DL	19