

# 09 Intégration

Bouchons et Mock

Thibaut HENIN

[www.arsouyes.org](http://www.arsouyes.org)

# Unit Test vs Integration Test

<https://www.youtube.com/watch?v=0GypdsJu1KE>



Quel est le problème ?

# Les tests unitaires

Quel cas d'utilisation ?

## Tester un composant

(Seul et isolé)

TP1 : Fonctions mathématiques

TP2 : Pile

TP3 : Fonction sur les dates

TP5 : Convertisseur markdown

TP6 : Traducteur

TP7 : Vérificateur de mot de passe

# Limite du test unitaire

TP9 : Contrôleur d'alarme

Plusieurs composants

(dépendants)



Quelles solutions ?

Bouchons

Tests d'intégration

Bouchons

# Qu'est-ce qu'un bouchon ?

Exemple dans le TP9

**Simuler le composant**

(il suffit de donner le change)





# Qu'est-ce qu'un « mock » ?

Exemple dans le TP9

## Bouchon

(il simule le composant)

## Surveille les appels

(il contrôle son utilisation)

Tests d'intégration

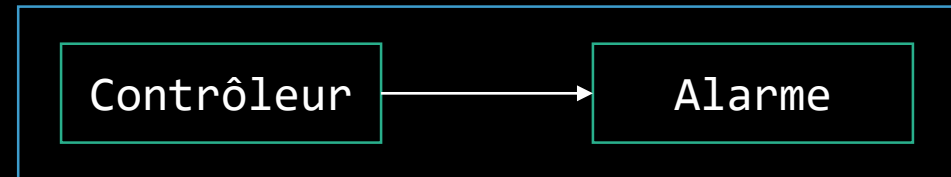
# Principe général

Test en groupe

(plusieurs composants)

Tester les interfaces

(le comportement individuel  
est déjà testé unitairement)



# Différents niveaux

Groupes plus ou moins larges

Classes

Packages

Programmes

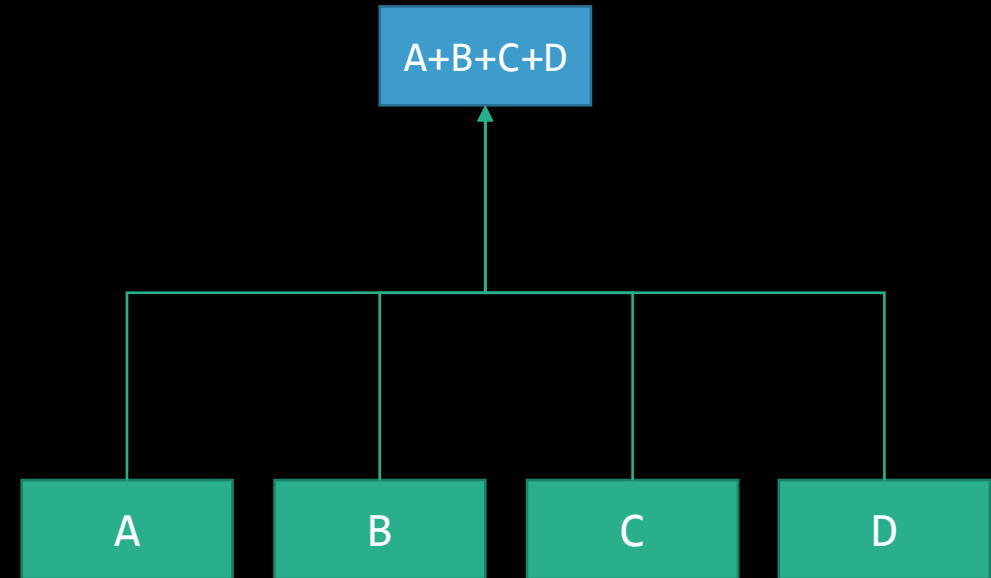
Matériels

Réseaux

Stratégie d'intégration

# Big Bang

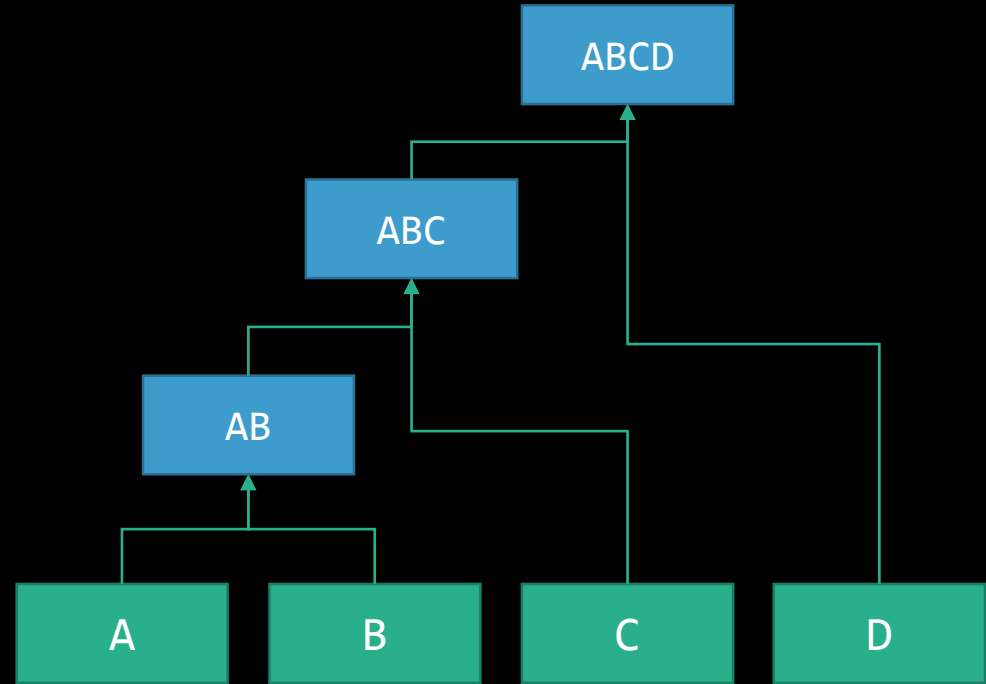
Intégration globale  
(Après avoir tout développé)



# Ascendant

Partir des dépendances

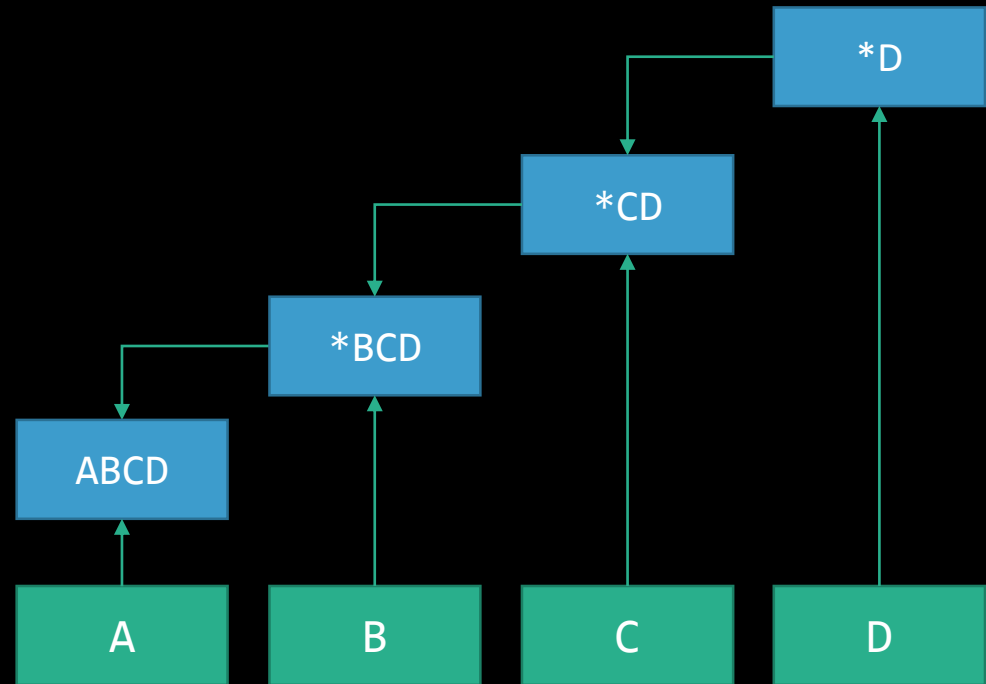
(tester et développer  
couche après couches)



# Descendant

Partir du global

(simuler les composants  
et les remplacer ensuite)

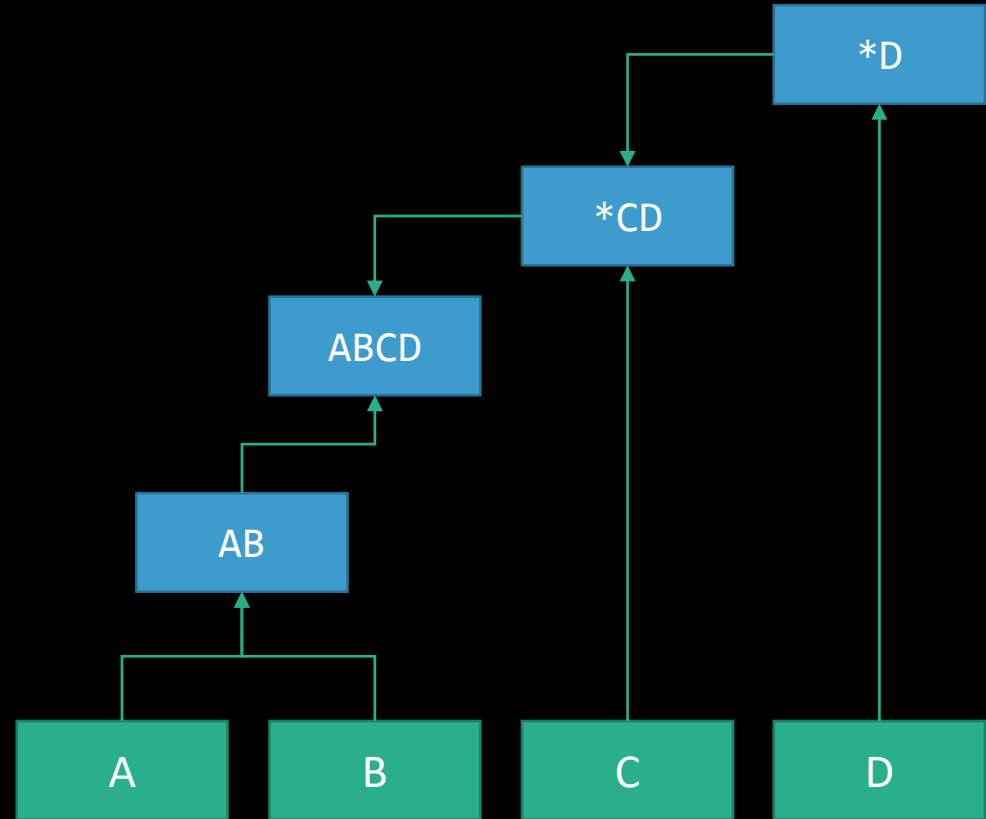




# Sandwich

Ascendant pour la base

Descendant pour le global



# Comment choisir ?

Tout dépend du contexte

Délais pour commencer

Délais pour terminer

Pilotes vs Bouchons

Possibilités de planifier

Two unit test suites.  
No integration testing.

© fasttime  
@ devrant

