

Tests unitaires (et tests de charge)

Animé par Sylvain Labasse

INTRODUCTION

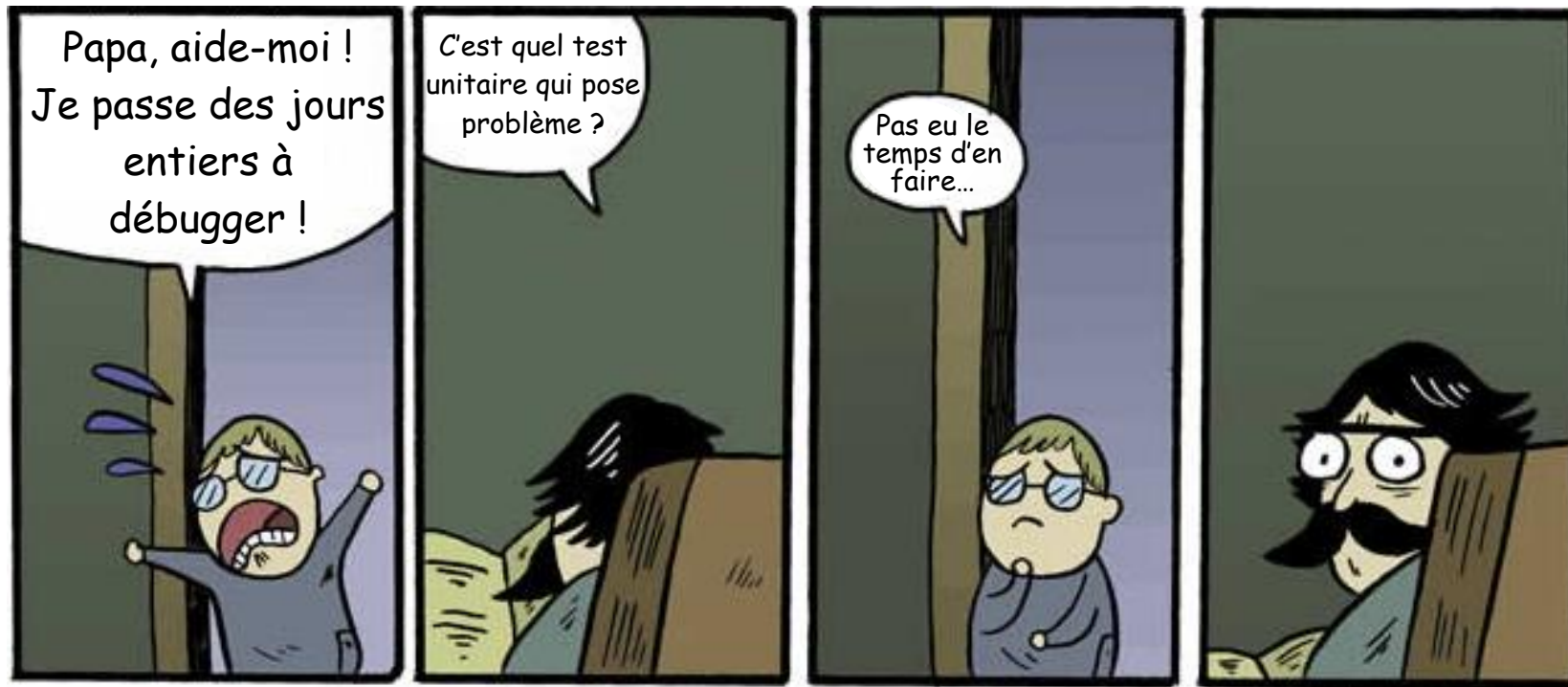


Figure 1 - src : <https://medium.com/globant/unit-testing-a-necessary-evil-9630322240e4>

EN FIN DE MODULE, VOUS SAUREZ...

Cibler les classes à tester unitairement

Identifier les dépendances d'une classe

Pratiquer l'inversion de contrôle (IoC)

Simuler les dépendances en préparant des stubs

Valider les interactions grâce à des mocks

PRE-REQUIS

Public

B3 - Développeur d'applications

Nécessaire

Notion de classe, d'héritage et polymorphisme
Syntaxes Python

DEROULEMENT

Support : <https://bit.ly/3VBFij5>

Copie des slides / Vidéos

Notes de cours

Ateliers

i majuscule

Evaluation

Ateliers sur 4

0	1	2	3	4
Non remis	Hors sujet	<	=	>
		attentes		

L'ENVIRONNEMENT

Matériel

PC sous Windows/Mac + Connexion Internet

Logiciel

IDE Python : PyCharm / VSCode

TESTS UNITAIRES ET TESTS DE CHARGE

Testabilité

Rappel des exigences

Dépendances

IoC et DI

Simulacres

Problématique

Prérequis

Stubs vs Mocks

Tests de charges

Principe

Outils

Locust

TESTABILITE

OBJECTIFS

Prise de conscience de l'impératif de testabilité

Identification et classification des dépendances

Traitements de découplage à appliquer

Conséquences pour l'application

TESTABILITE

→ Rappel des exigences

Dépendances

IoC et DI

Synthèse

RAPPEL DES EXIGENCES

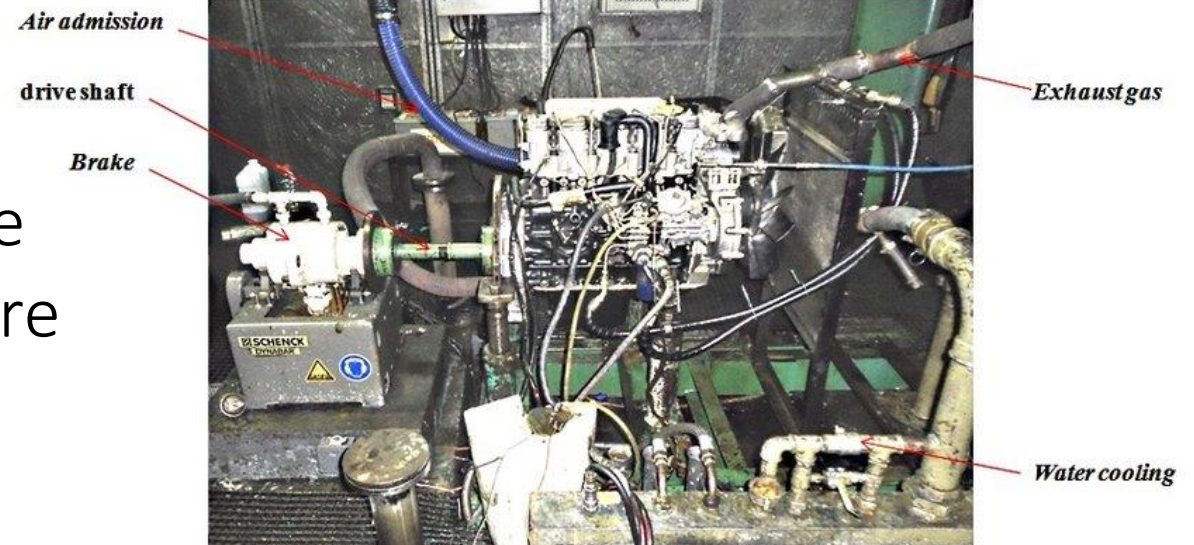
Banc de test

Validation d'une classe en boîte noire, qq soit l'utilisation
Projet/Rép. de test indépendant du projet, mais même dépôt
1 classe testée = 1 classe de test

Figure 2 - src : https://www.researchgate.net/figure/Le-banc-dessai-utilise-pour-la-conduite-des-essais-moteur-4_fig1_335568309

Critères

Autonome et déterministe
Facile à écrire et facile à lire
Exécution rapide



TESTABILITE

✓ Rappel des exigences

➔ Dépendances

IoC et DI

Synthèse

DEPENDANCES

Définition

Toute fonction ou classe extérieure à une classe utilisée dans son code.

Exemples

Classe : paramètre, champ, singleton

Instructions non déterministes : Aléatoire, heure, ...

Interactions externes : Persistance, API, UI, ...

TESTABILITE

- ✓ Rappel des exigences

- ✓ Dépendances

- ➔ IoC et DI

Synthèse

INVERSION DE CONTROLE (IoC)

A utilise la méthode g() de B

Création d'un interface avec g()

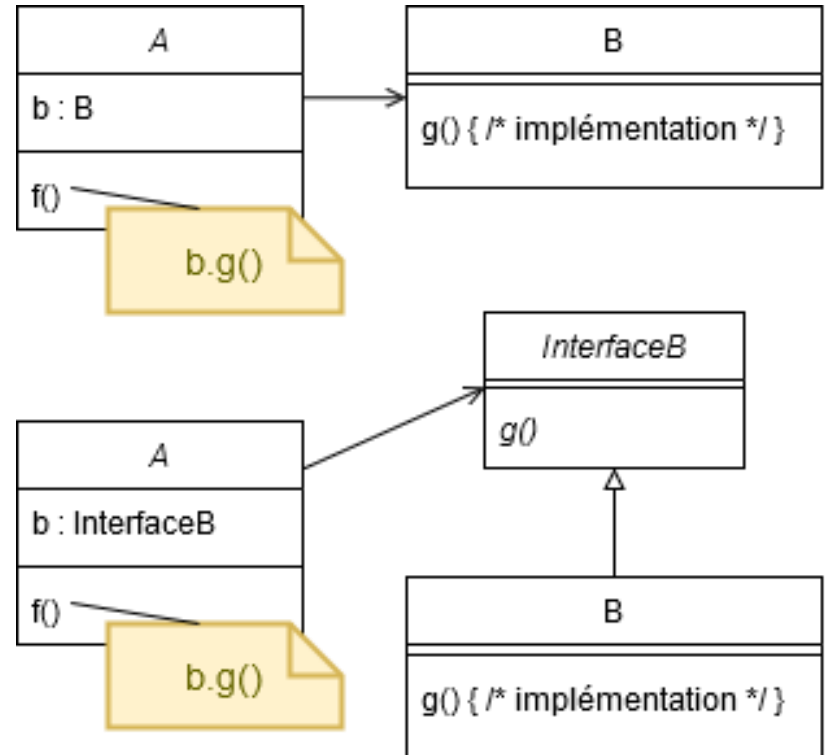
A ne connaît plus B mais l'interface

Conséquences : soLID

A est testable sans B

A ne peut (directement) instancier B

Il faudra « juste » simuler B



INJECTION DE DEPENDANCE (DI)

Principe

Mécanisme de mise à disposition des implémentations.
Distribue les interfaces requises par les classes.

Solutions

Solution « maison » : Dictionnaire, classe conteneur, ...
Framework/Librairie d'injection de dépendance

TESTABILITE

- ✓ Rappel des exigences
- ✓ Dépendances
- ✓ IoC et DI
- ➔ Synthèse

RESUME

Prise de conscience de l'impératif de testabilité

Identification et classification des dépendances

Traitements de découplage à appliquer

Conséquences pour l'application