

# Conception du logiciel d'aide à la création de livres dont vous êtes le héros (LDVH)

---

Par la compagnie **STLoosers** dans le cadre de l'UE d'Ingénierie Logicielle

Daniel Sima  
Olando Bazil  
Quentin Cavoille  
Arthur Pendellou

Zeid Fazazi  
Abdelkader Boumessaoud  
Farouck Cherfi  
Amaury Curiel

# Sommaire

I. Diagramme des composants

II. Diagramme de structure interne

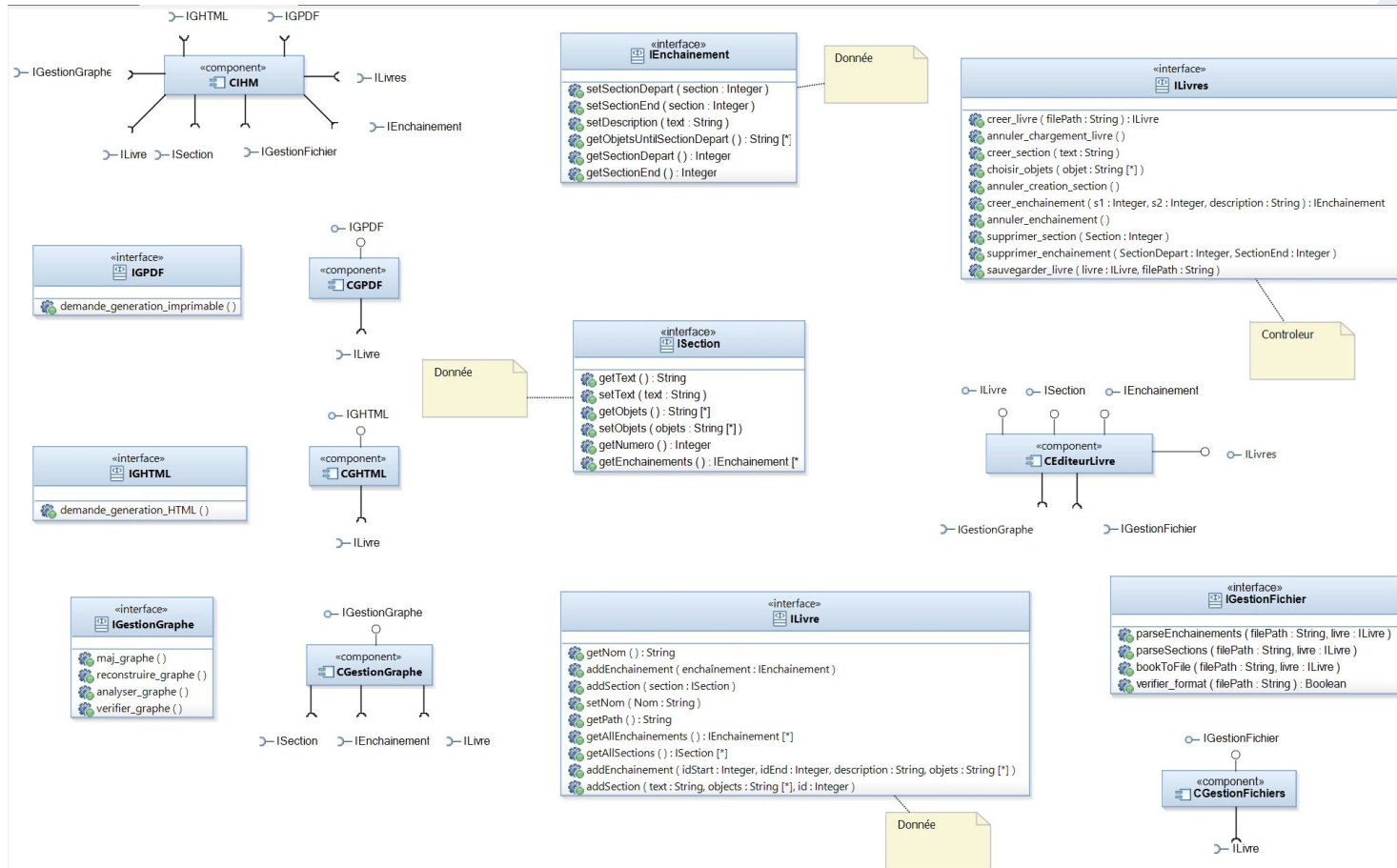
III. Charger un livre

- A. Diagramme de séquence composant
- B. Diagramme de conception détaillée
- C. Test d'intégration

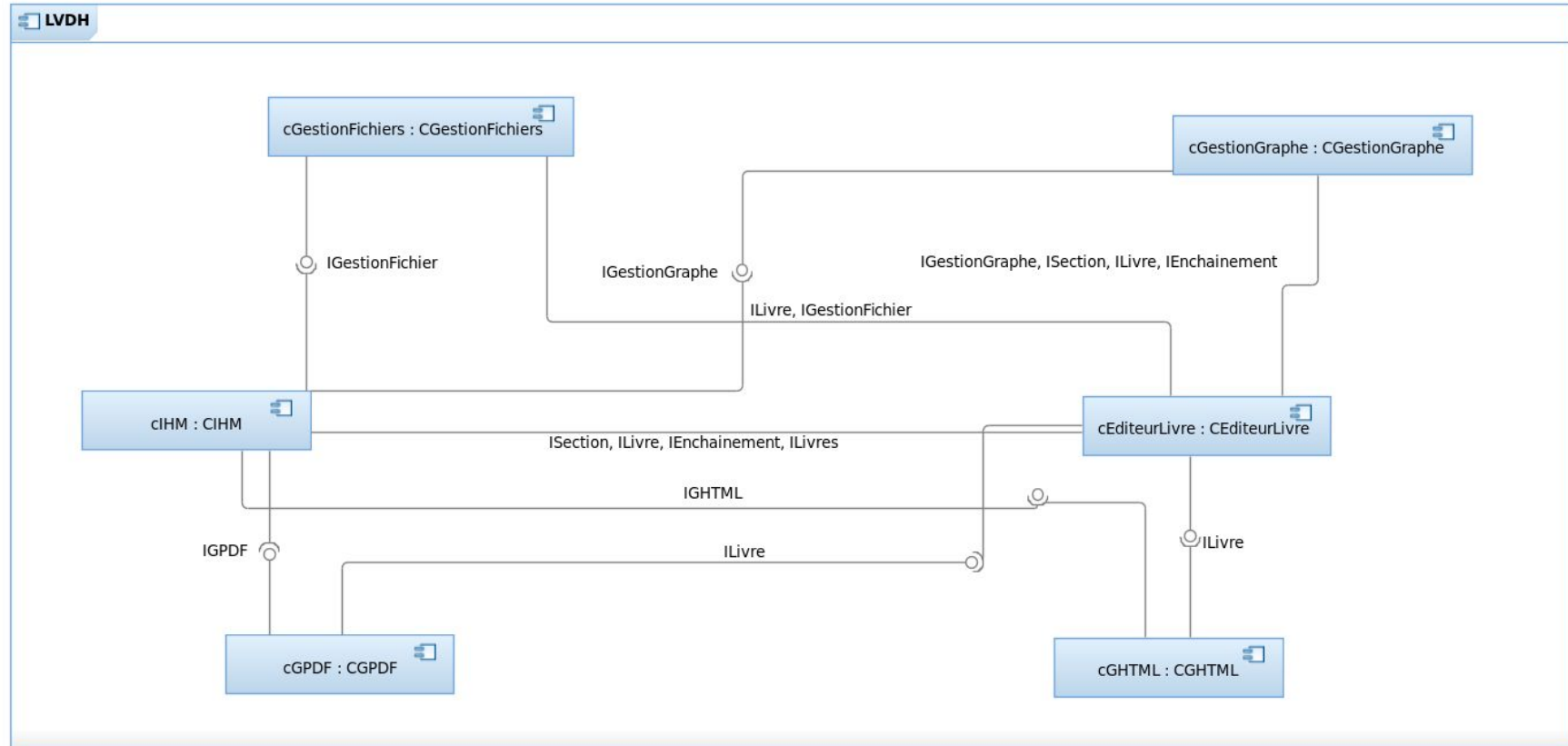
IV. Définir un nouvel enchaînement

- A. Diagramme de séquence composant
- B. Diagramme de conception détaillée
- C. Test d'intégration

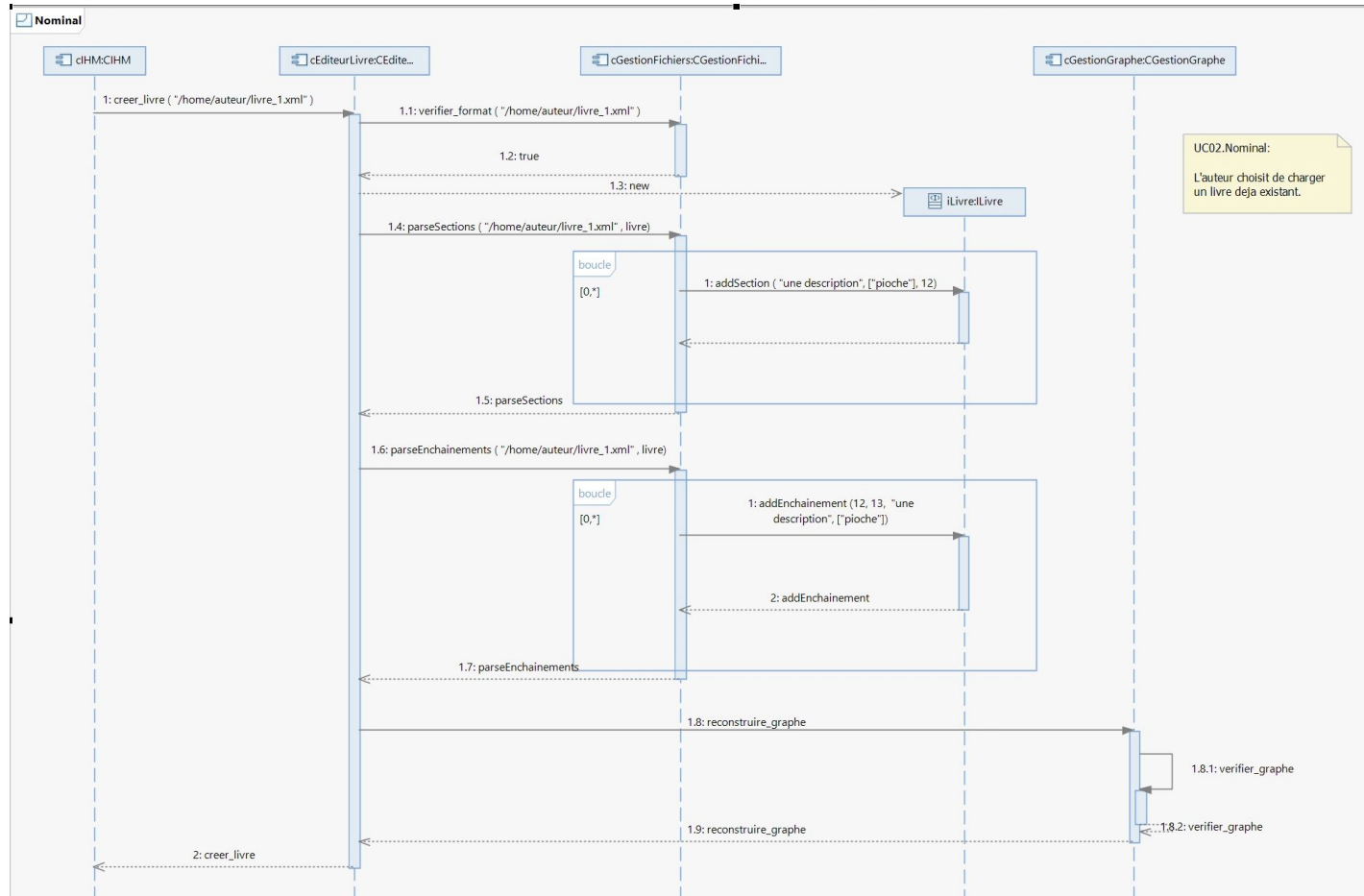
# I. Diagramme de composants



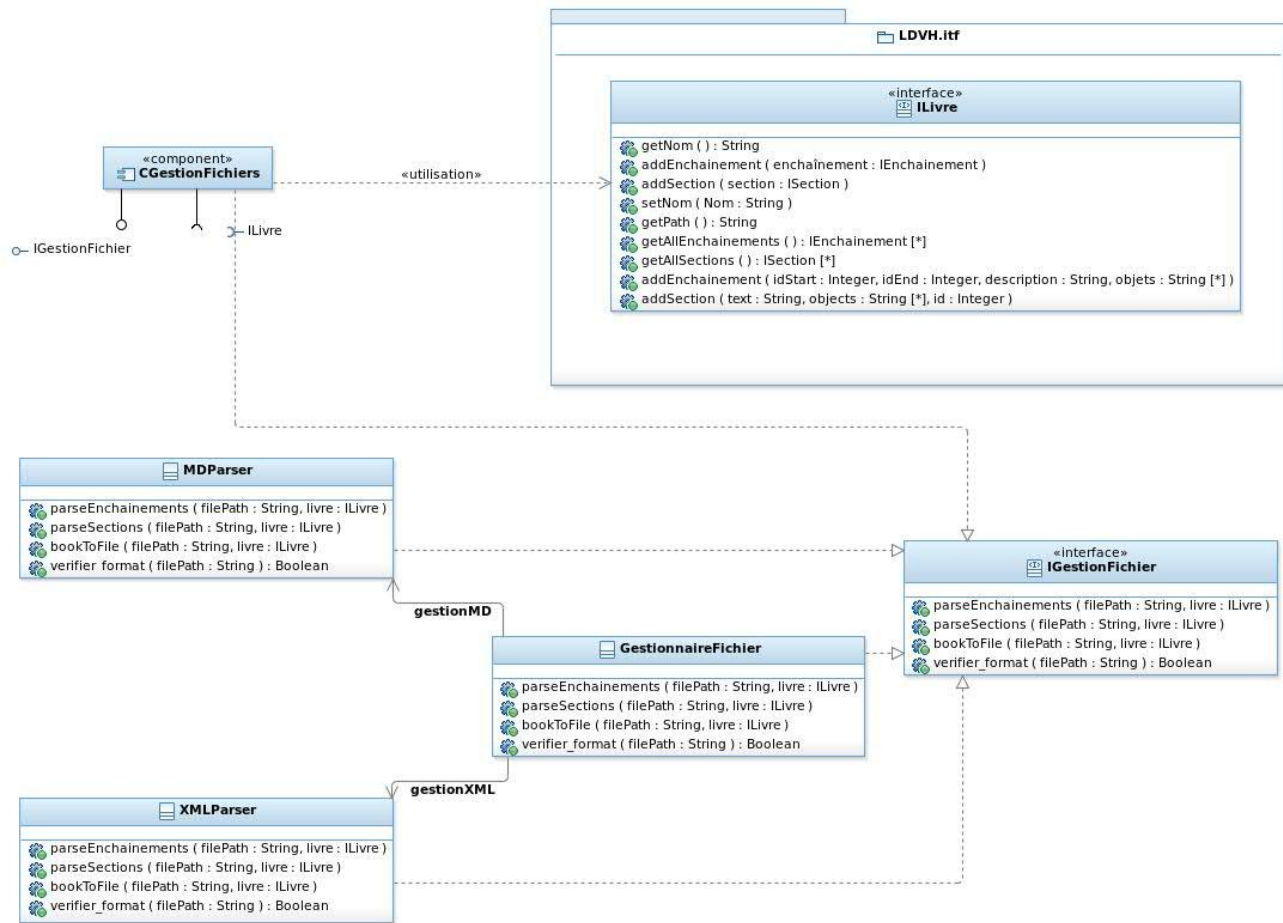
## II. Diagramme de structure interne



# III. Charger livre - Diagramme de séquence composant



## III. Charger livre - Diagramme de conception détaillée



## III. Charger livre - Test d'intégration

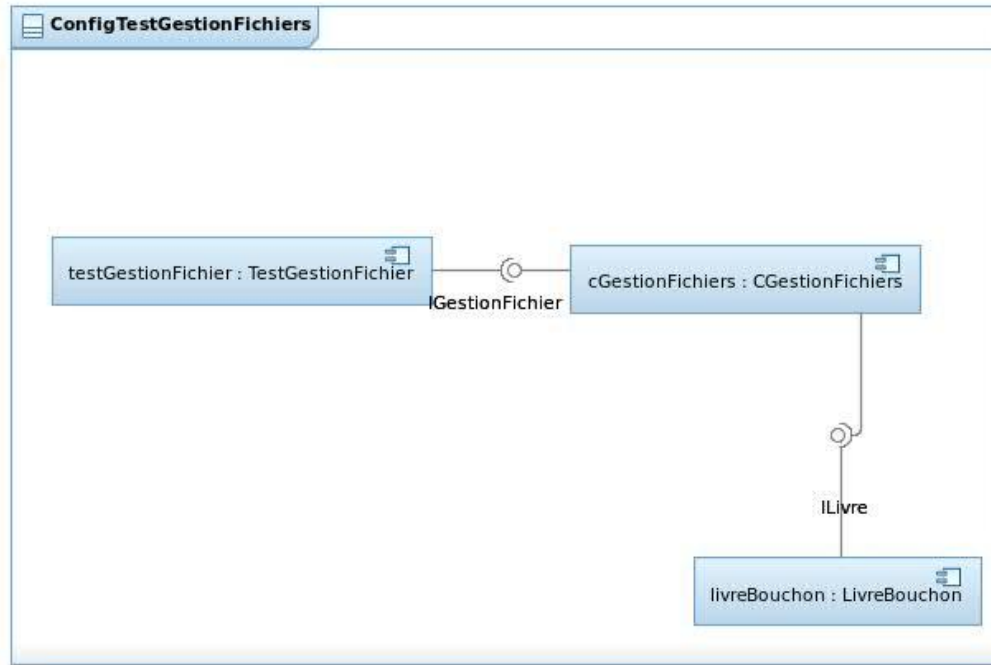


Diagramme de structure interne des configurations de test d'intégration des composants

## III. Charger livre – Test d'intégration

### Contexte:

- Livre : "Livre"
- Sections de "Livre": {1; 2; 3}
- Enchaînements existants: (1, 2, "ouvre porte A")  
(2, 3, "ouvre porte B")

### Objectif du test:

Tester le chargement du livre existant "Livre"

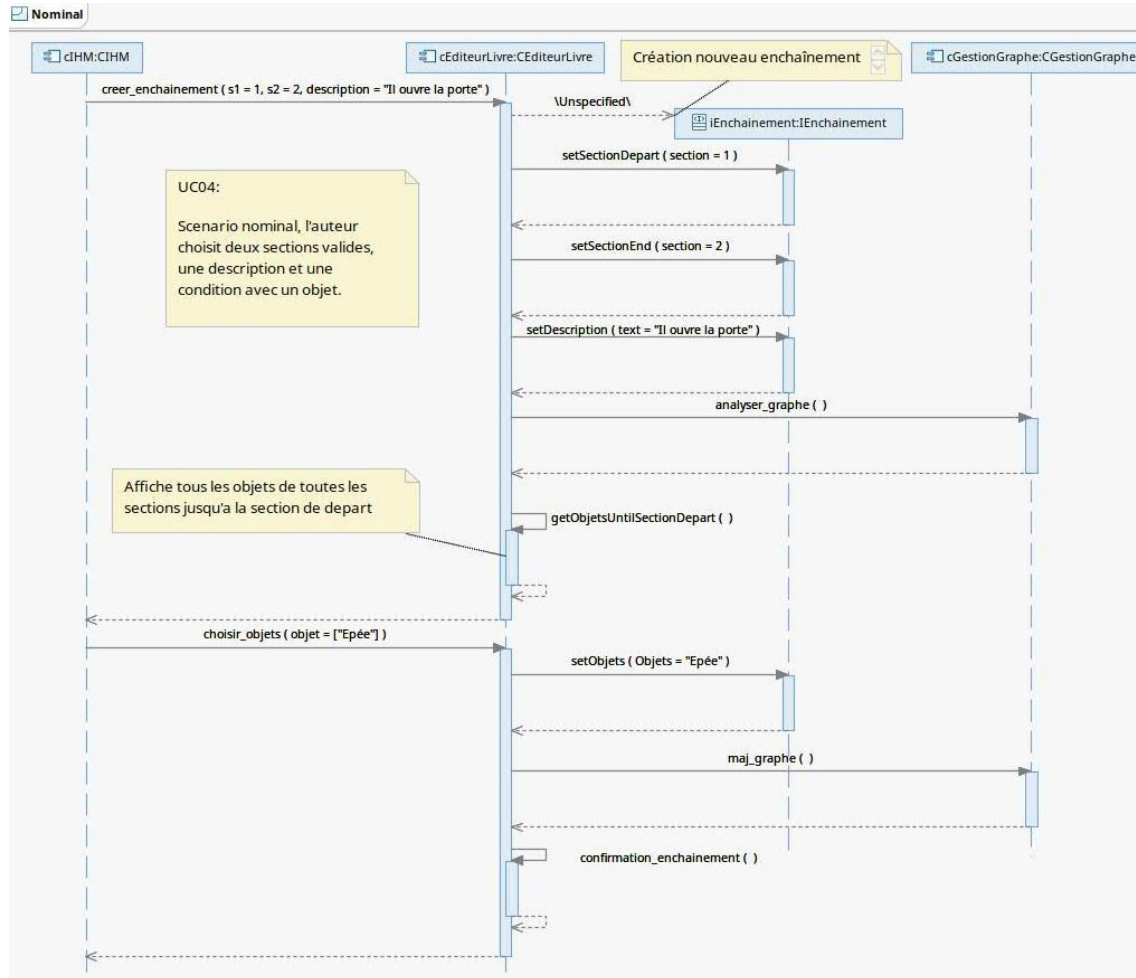
### Tests:

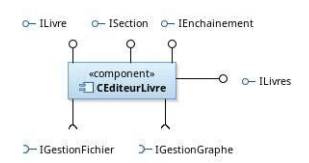
- **assertEquals**(creer\_livre("/home/auteur/livre\_1.xml").getPath(), "/home/auteur/livre\_1.xml");  
#True
- **assertEquals**(creer\_livre("/home/auteur/livre\_1.xml").getAllSections()[0].getNumero(), 1) #True
- **assertEquals**(creer\_livre("/home/auteur/livre\_1.xml").getAllSections()[1].getNumero(), 2) #True
- **assertEquals**(creer\_livre("/home/auteur/livre\_1.xml").getAllSections()[2].getNumero(), 3) #True
- **assertEquals**(creer\_livre("/home/auteur/livre\_1.xml").getAllEnchainements()[0].getDescription(),  
"ouvre porte A") # True
- **assertEquals**(creer\_livre("/home/auteur/livre\_1.xml").getAllEnchainements()[1].getDescription(),  
ouvre porte B") # True



# IV. Définir un nouvel enchaînement - Diagrammes de séquence

## composant





## IV. Définir un nouvel enchaînement – Test d'intégration

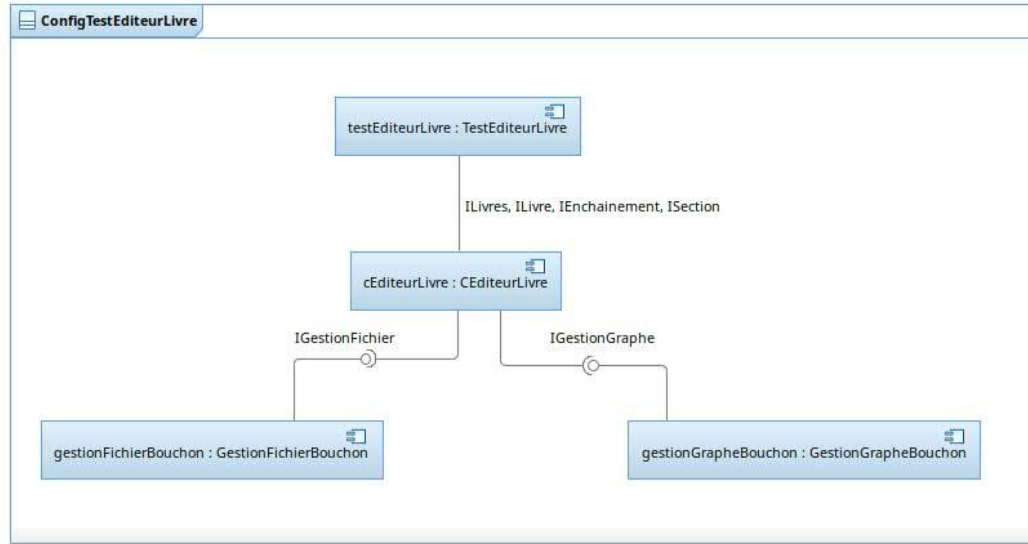


Diagramme de structure interne des configurations de test d'intégration des composants

## IV. Définir un nouvel enchaînement - Test d'intégration

### Contexte:

- Livre : "Livre"
- Sections de "Livre": {1; 2; 3; 4}
- Enchaînements existants: (1, 3, "allume une bougie")

### Objectif du test:

Tester la création d'un nouvel enchaînement entre les sections 1 et 2 avec la description "Ouvre la porte"

### Tests:

- **assertEquals**(creer\_enchainement(1, 2, "il ouvre la porte")).getSectionEnd(), 2); #True
- **assertExists**(creer\_enchainement(1, 2, "il ouvre la porte"), Livre.getAllEnchainements()) #True
- **assertEquals**(creer\_enchainement(4, 5, "description") in L.getAllEnchainements(), false) # True  
(cas d'erreur)

**Merci pour votre attention !**