

# UML2

## Diagrammes de composants

Réf. "Introduction à UML2" Miles, Hamilton. Ed. O'Reilly. 2006

# Diagrammes de composants

Développement logiciel :

Découpage en "gros" éléments

Architecture globale du système

→ gérer la complexité

→ gérer les dépendances

→ Composants

(morceaux logiciels + faciles à gérer)

# Diagrammes de composants

## Composants

- éléments encapsulés, réutilisables
- remplaçables, échangeables
- combinables, "briques"
- taille : 1 classe → 1 sous-système

## Ex.

enregistreur d'événements, parseur XML, caddie ...

# Diagrammes de composants

## Un composant

- fournit des services
- utilise des données / d'autres services
- est associé à des classes / à d'autres composants
- implémente des interfaces
- possède des opérations
- a une structure interne

# Diagrammes de composants

## Composant vs Classe

Classe Utilisateur

informations (nom, adresse, ...)

Composant GestionUtilisateurs

création des comptes, vérification de l'authenticité, ...

1 composant : + de responsabilités

→ contient des classes, utilise d'autres composants

# Diagrammes de composants

**Couplage** entre composants (dépendance)

Couplage le + faible possible

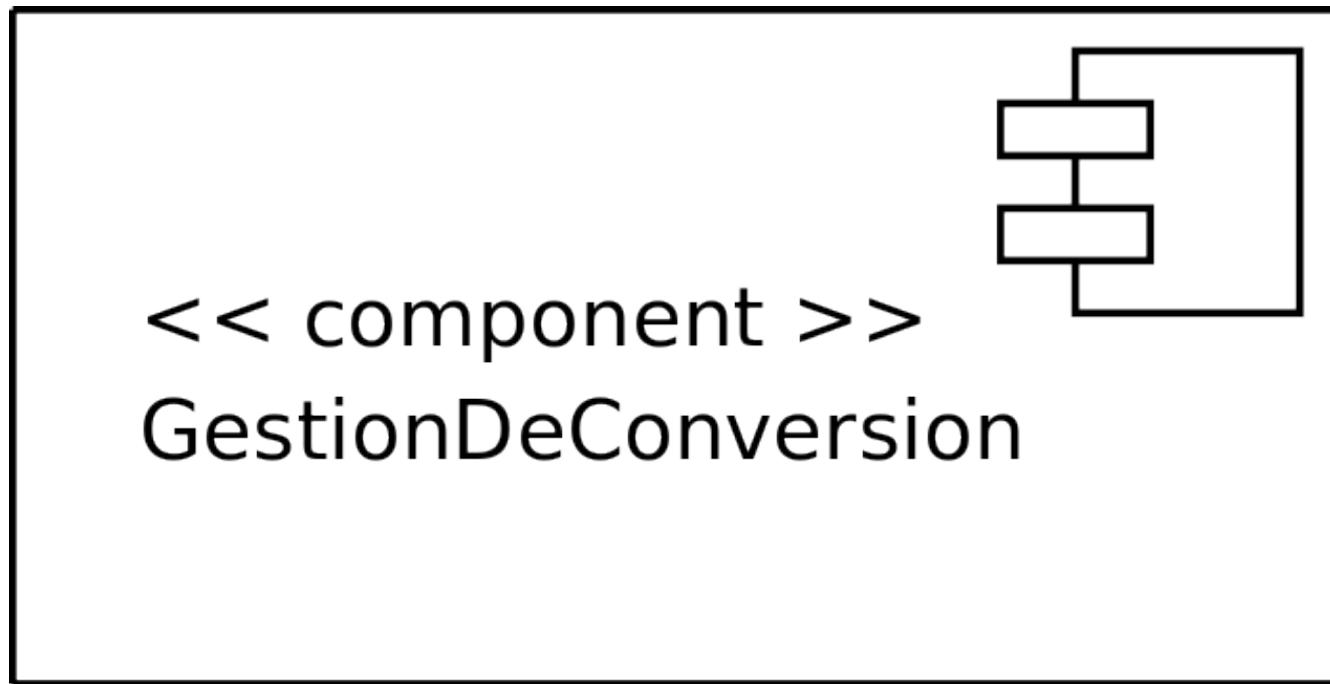
→ composants "autonomes"

Utilisation d'interfaces

→ peu/pas d'effet sur le reste du système en cas  
de modification interne

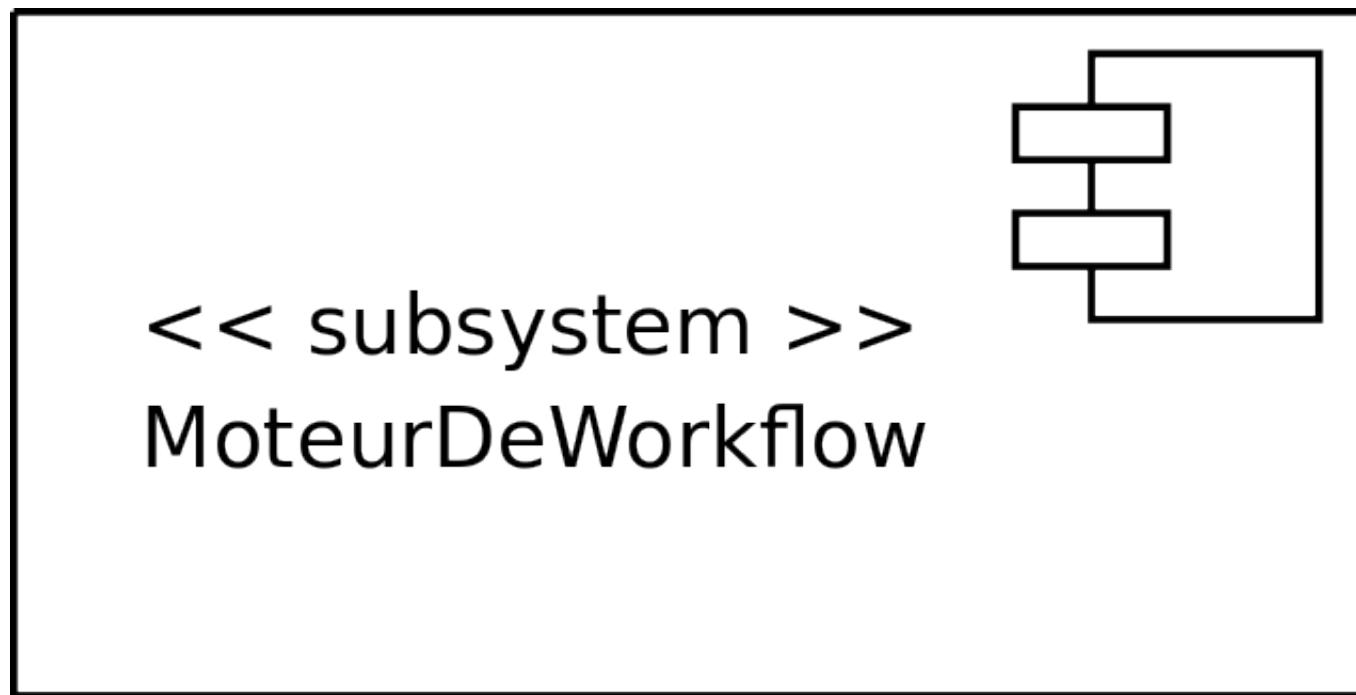
# Diagrammes de composants

Représentation d'un composant



# Diagrammes de composants

Représentation d'un "gros" composant



# Diagrammes de composants

Interactions au travers des interfaces

→ faible couplage

Interface fournie

mise en œuvre

service offert

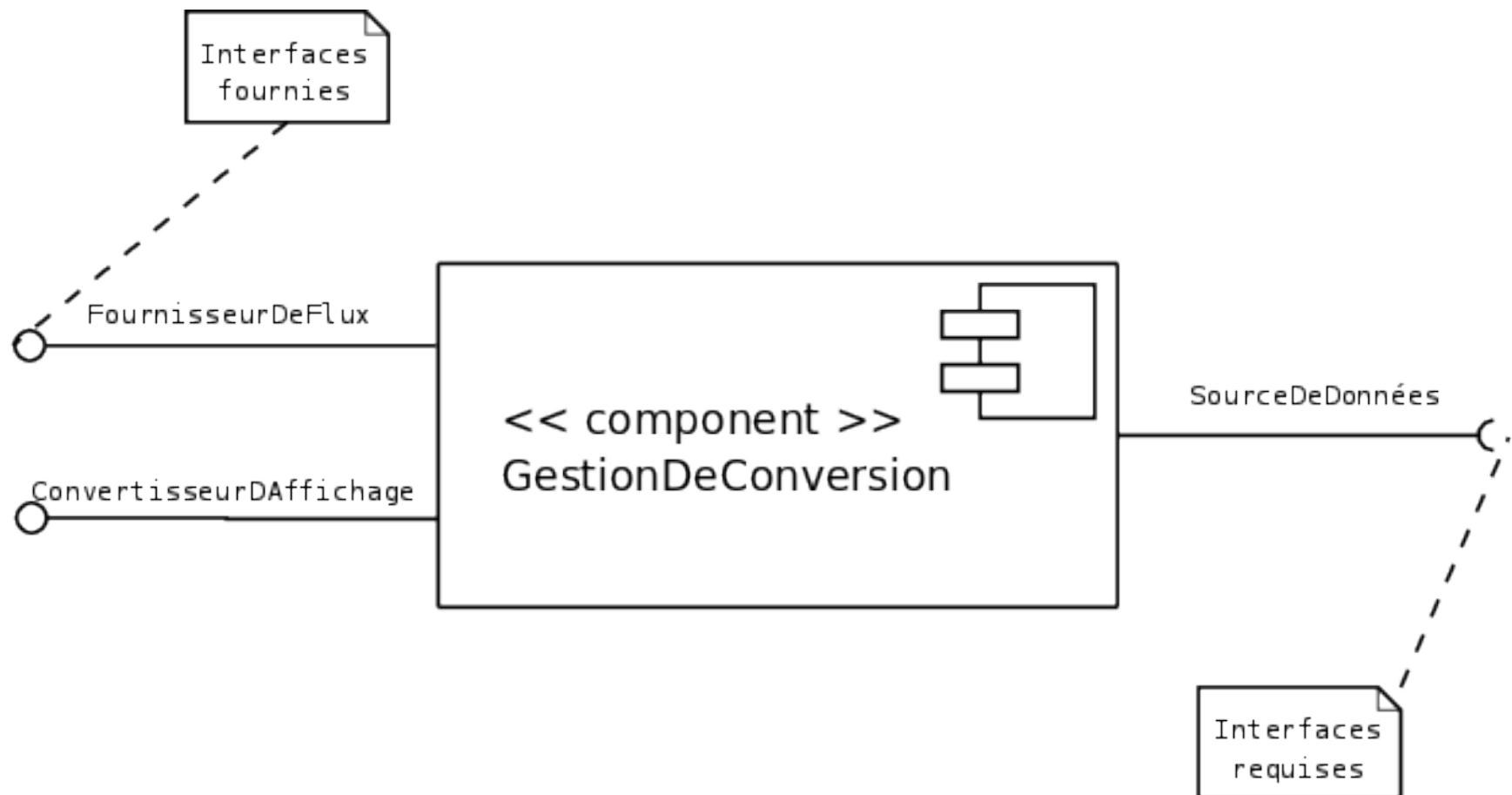
Interface requise

besoin pour fonctionner

classe ou composant (via une interface)

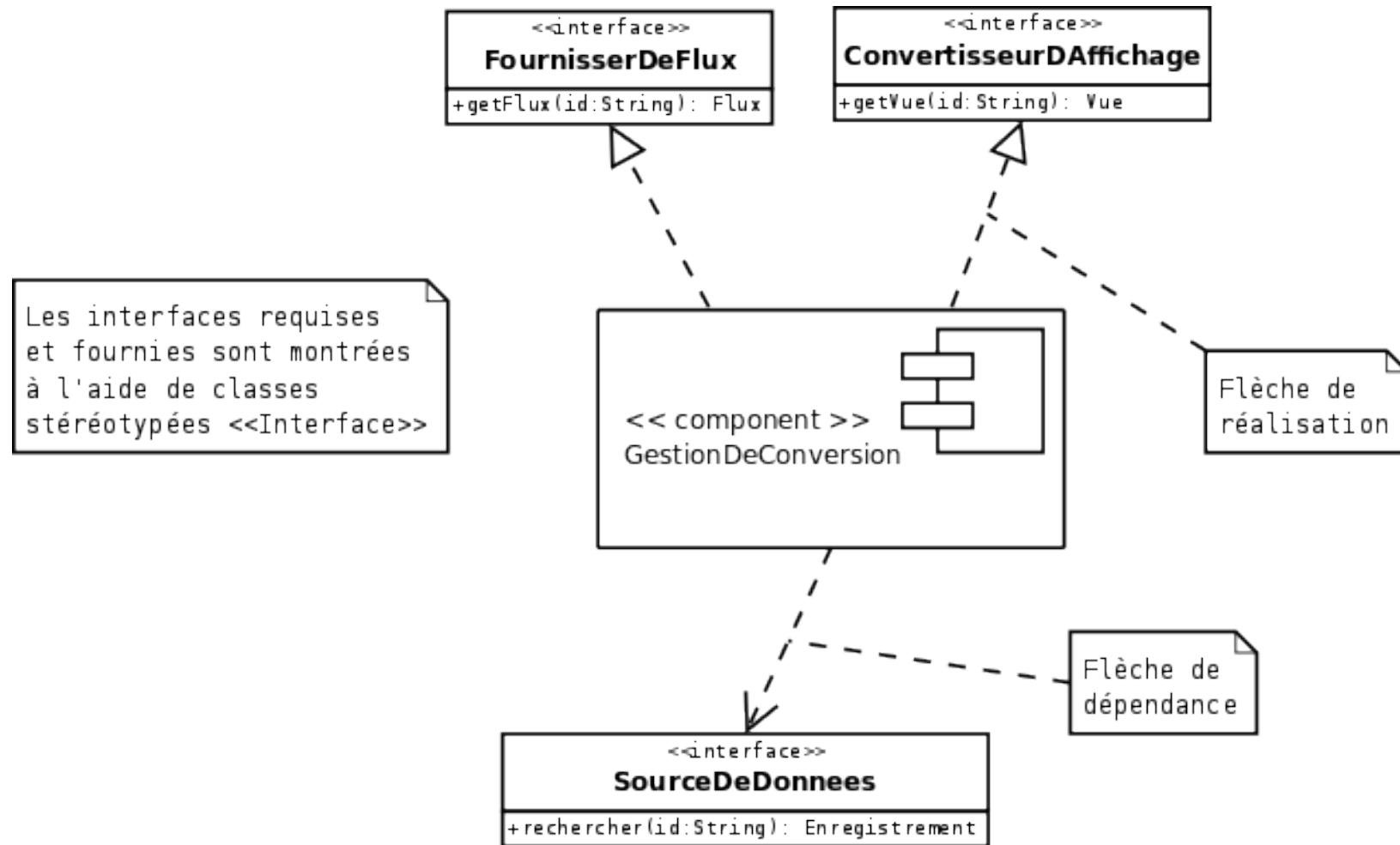
# Diagrammes de composants

## Représentation des interfaces



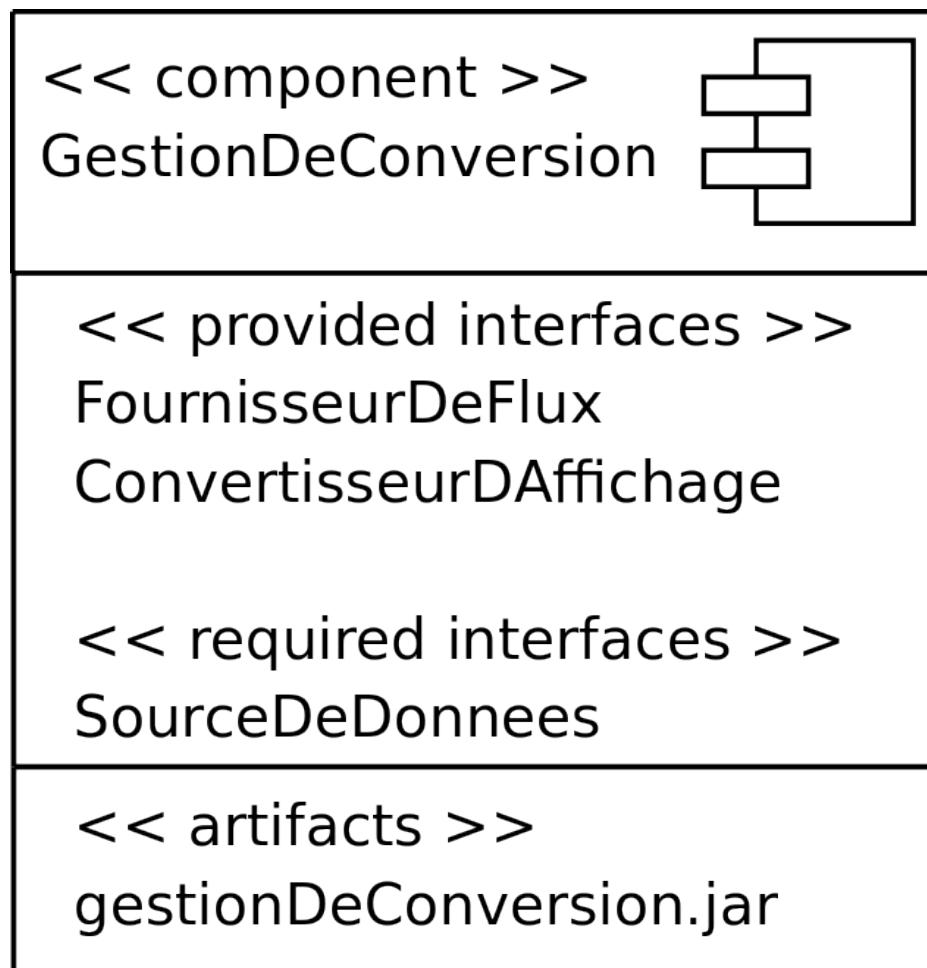
# Diagrammes de composants

## Représentation des interfaces (et opérations)



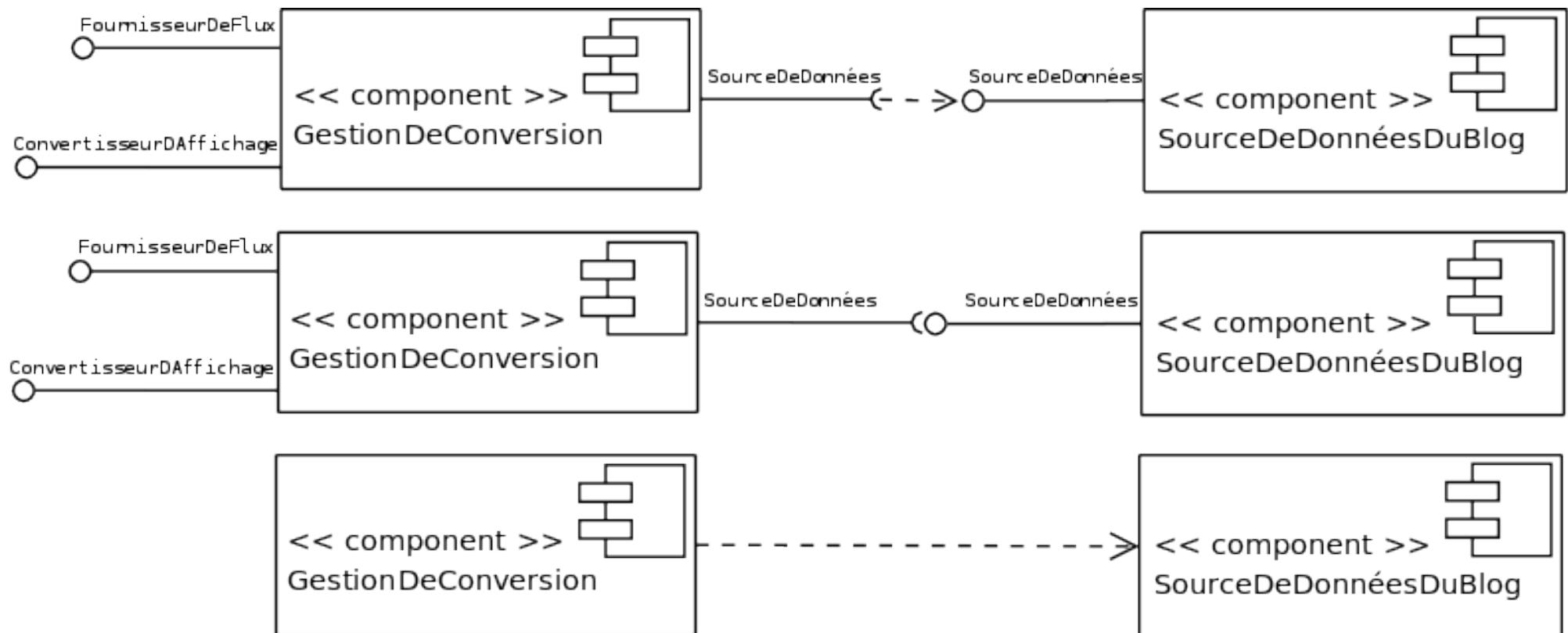
# Diagrammes de composants

Représentation des interfaces ( + compacte )



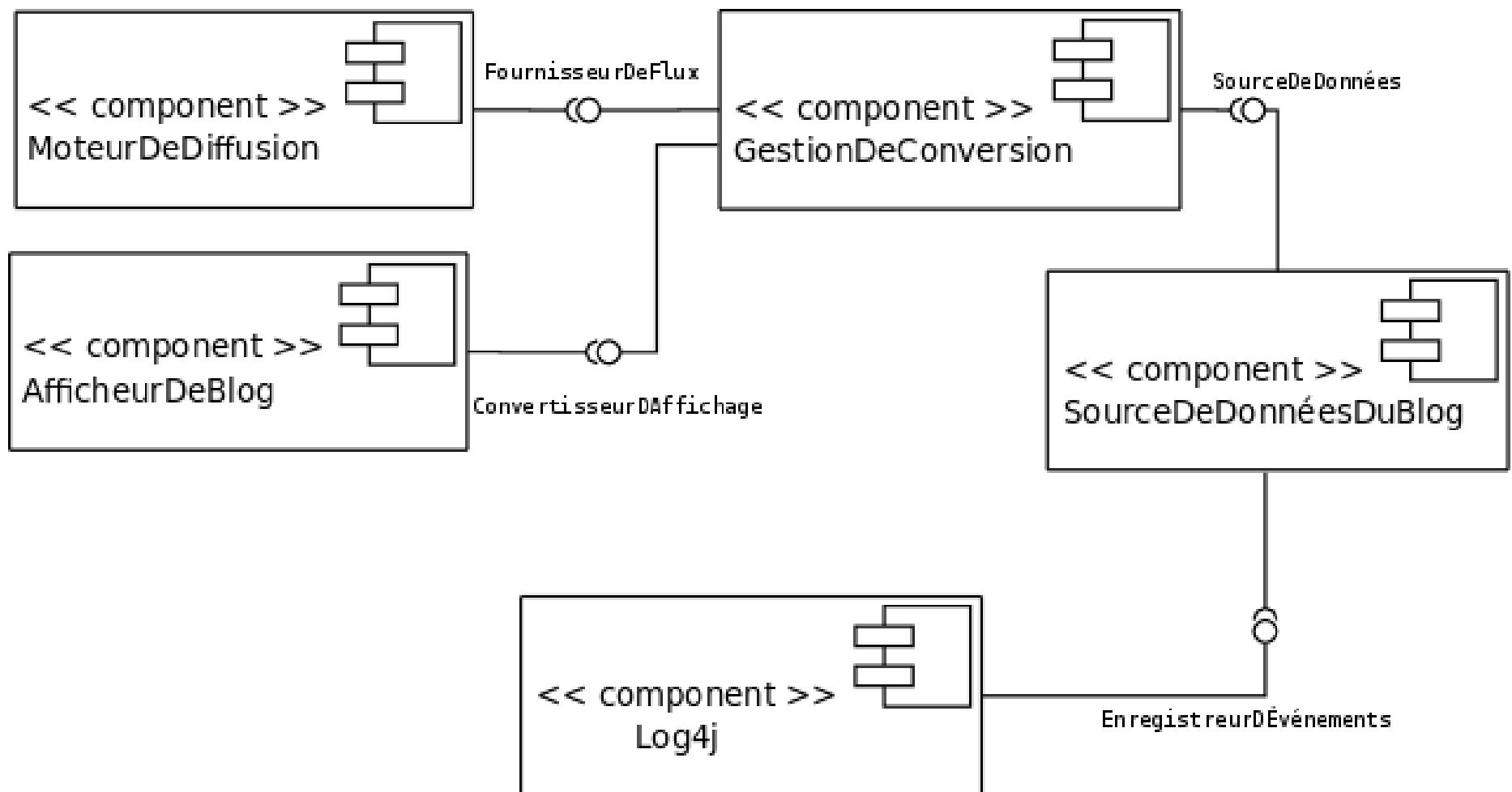
# Diagrammes de composants

## Coopération des composants



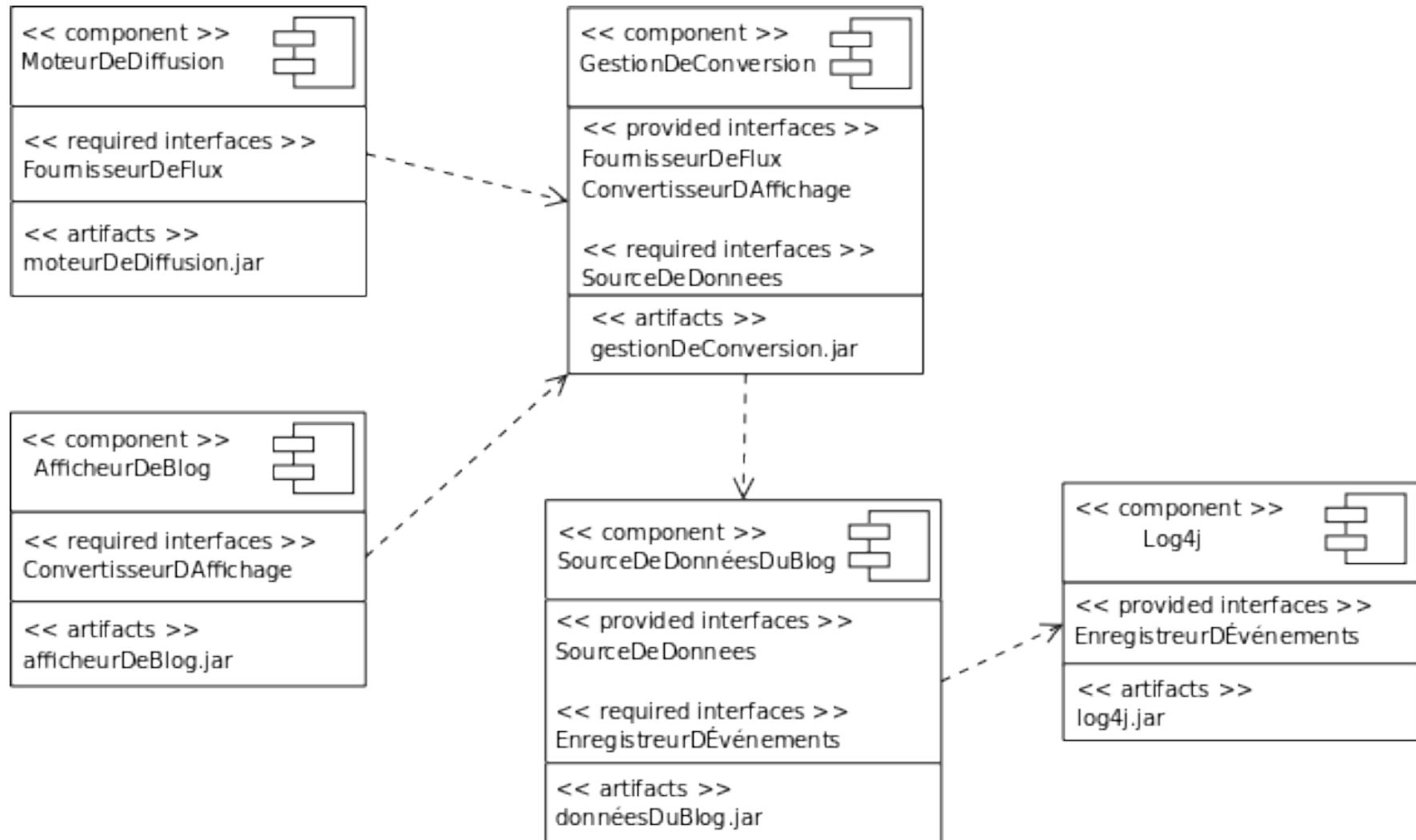
# Diagrammes de composants

## Vue de l'architecture système (interfaces)



# Diagrammes de composants

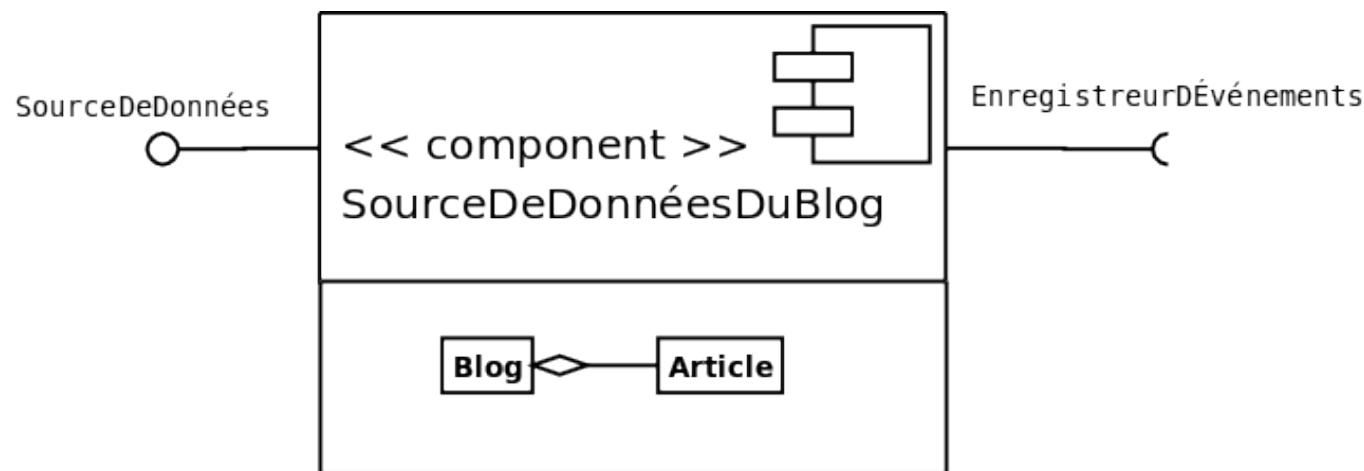
## Vue de l'architecture système (dépendances)



# Diagrammes de composants

## Classes réalisant un composant

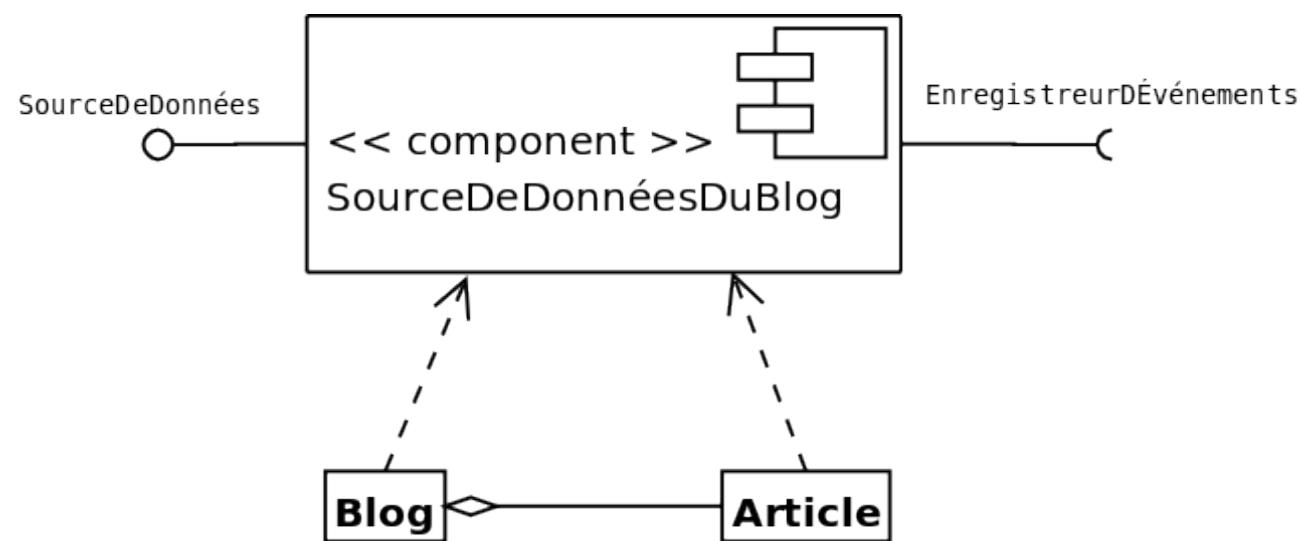
- classes nécessaires
- utilisées dans la mise œuvre
- "contenues" dans le composant



# Diagrammes de composants

## Représentation alternatives de la réalisation

<< component >>	SourceDeDonnéesDuBlog	
<< provided interfaces >>	SourceDeDonnées	
<< required interfaces >>	EnregistreurDÉvénements	
<< realizations >>	Blog	
	Article	
<< artifacts >>	gestionDeConversion.jar	



# Diagrammes de composants

## Ports d'un composant

"manières" d'utiliser un composant

chaque port peut fournir / nécessiter des interfaces

## Structure interne d'un composant

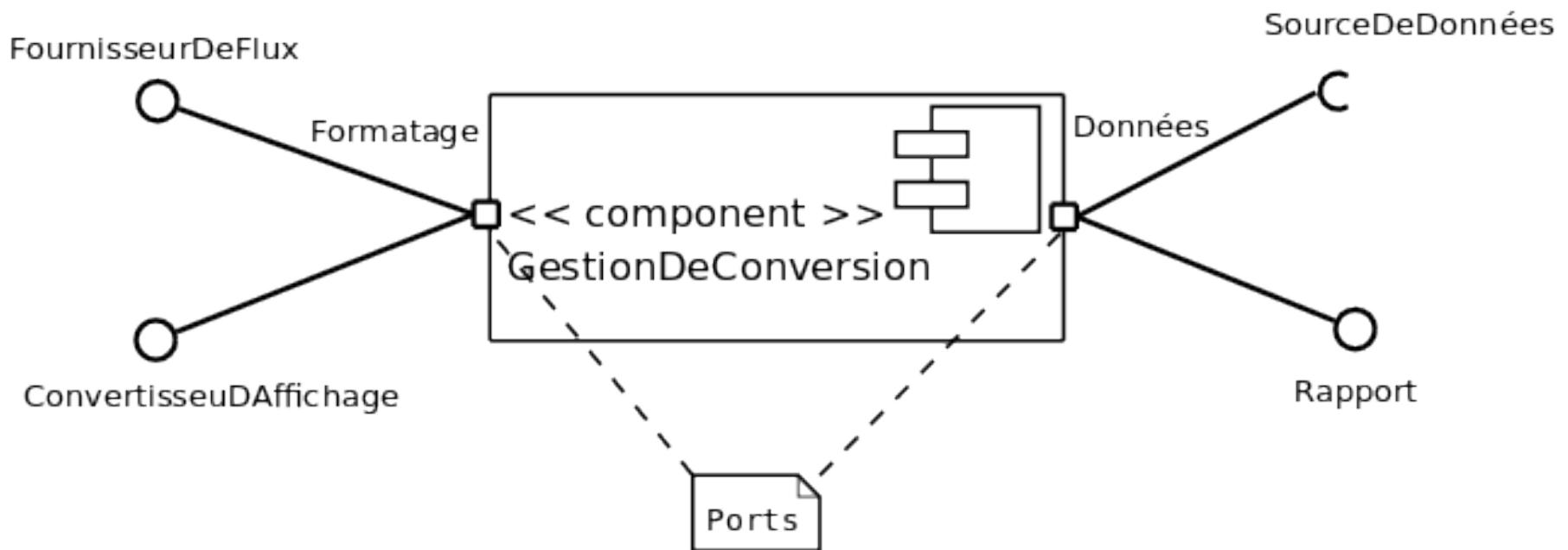
"parties" du composant

propriétés

connecteurs

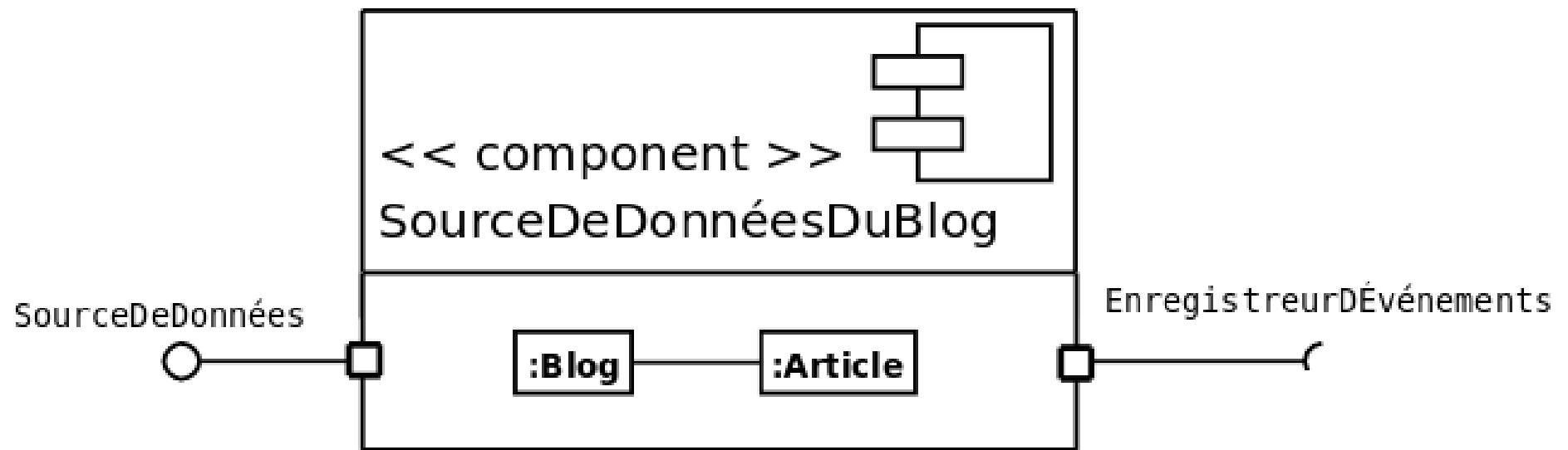
# Diagrammes de composants

## Ports d'un composant



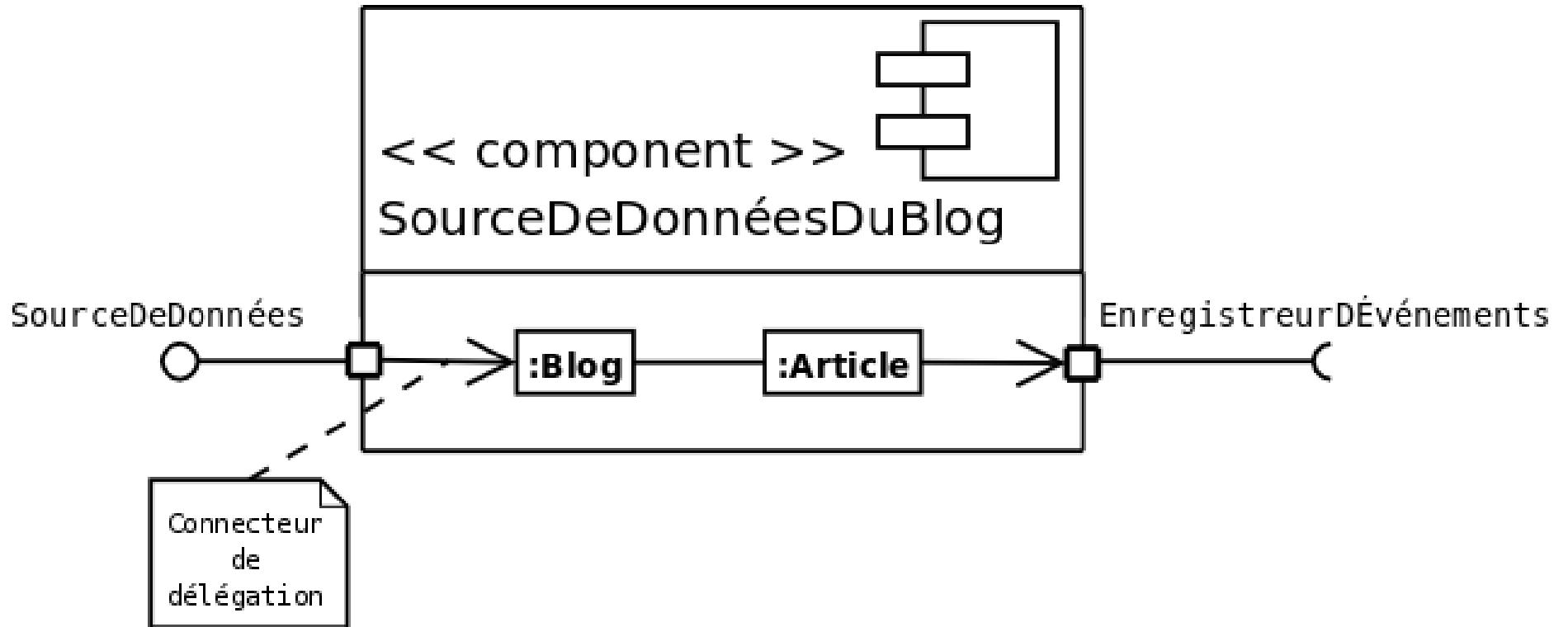
# Diagrammes de composants

## Structure interne



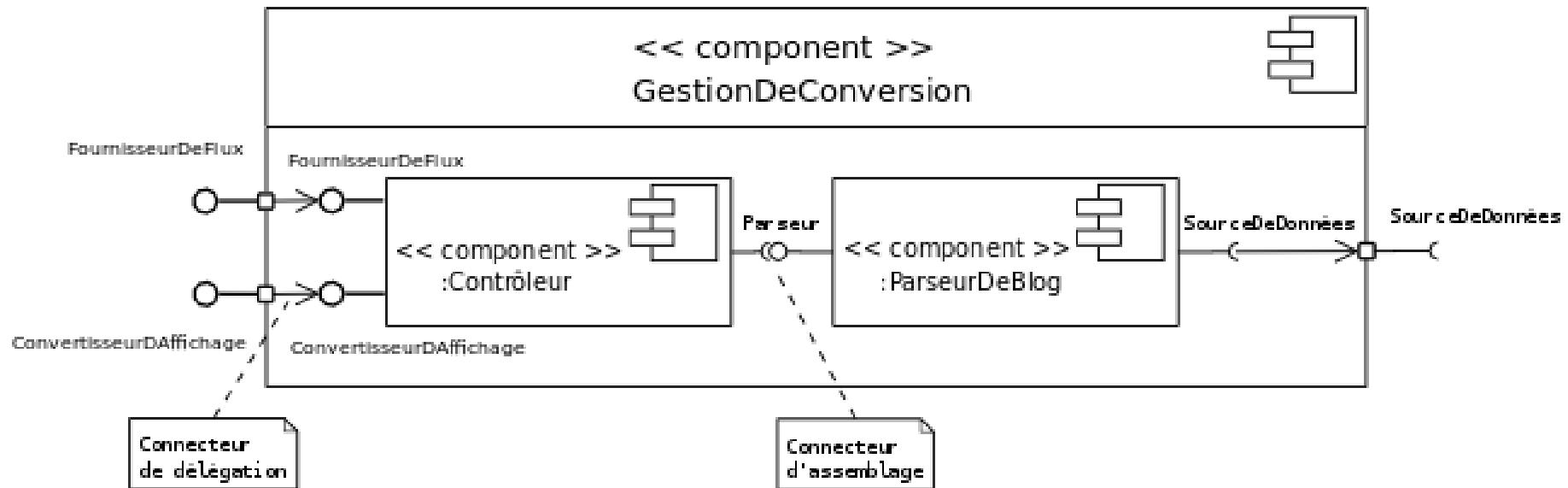
# Diagrammes de composants

Interface réalisée et interface requise



# Diagrammes de composants

Interface réalisée et interface requise  
( en montrant les interfaces internes )



# Diagrammes de composants

## Boîte noire vs boîte blanche

