



## Chapitre 2 :

# Architecture et Fonctions d'un Système Relationnel de Base de Données

# Introduction

- Un des principaux avantages du modèle relationnel est l'indépendance des données par rapport aux supports physiques et aux chemins d'accès disponibles. Un autre avantage est la possibilité de définir à tout moment dans le cycle de vie de la base de données, de nouvelles relations liées ou non, à celles déjà existantes.



# Introduction

- Un des principaux avantages du modèle relationnel est l'indépendance des données par rapport aux supports physiques et aux chemins d'accès disponibles. Un autre avantage est la possibilité de définir à tout moment dans le cycle de vie de la base de données, de nouvelles relations liées ou non, à celles déjà existantes.



# Introduction

- Un des principaux avantages du modèle relationnel est l'indépendance des données par rapport aux supports physiques et aux chemins d'accès disponibles. Un autre avantage est la possibilité de définir à tout moment dans le cycle de vie de la base de données, de nouvelles relations liées ou non, à celles déjà existantes.



# Introduction

- Un des principaux avantages du modèle relationnel est l'indépendance des données par rapport aux supports physiques et aux chemins d'accès disponibles. Un autre avantage est la possibilité de définir à tout moment dans le cycle de vie de la base de données, de nouvelles relations liées ou non, à celles déjà existantes.
- De plus, une amélioration appréciable a été apportée dans le domaine **de l'intégrité et la sûreté de fonctionnement** et dans **l'utilisation** des bases de données.



# Introduction

- Dans ce chapitre les différents aspects de la technologie introduits pour réaliser ces avantages sont passés en revue que nous illustrons par des exemples d'implémentation dans le système Oracles ou dans l'un des **premiers systèmes** développé par IBM appelé **System R**.



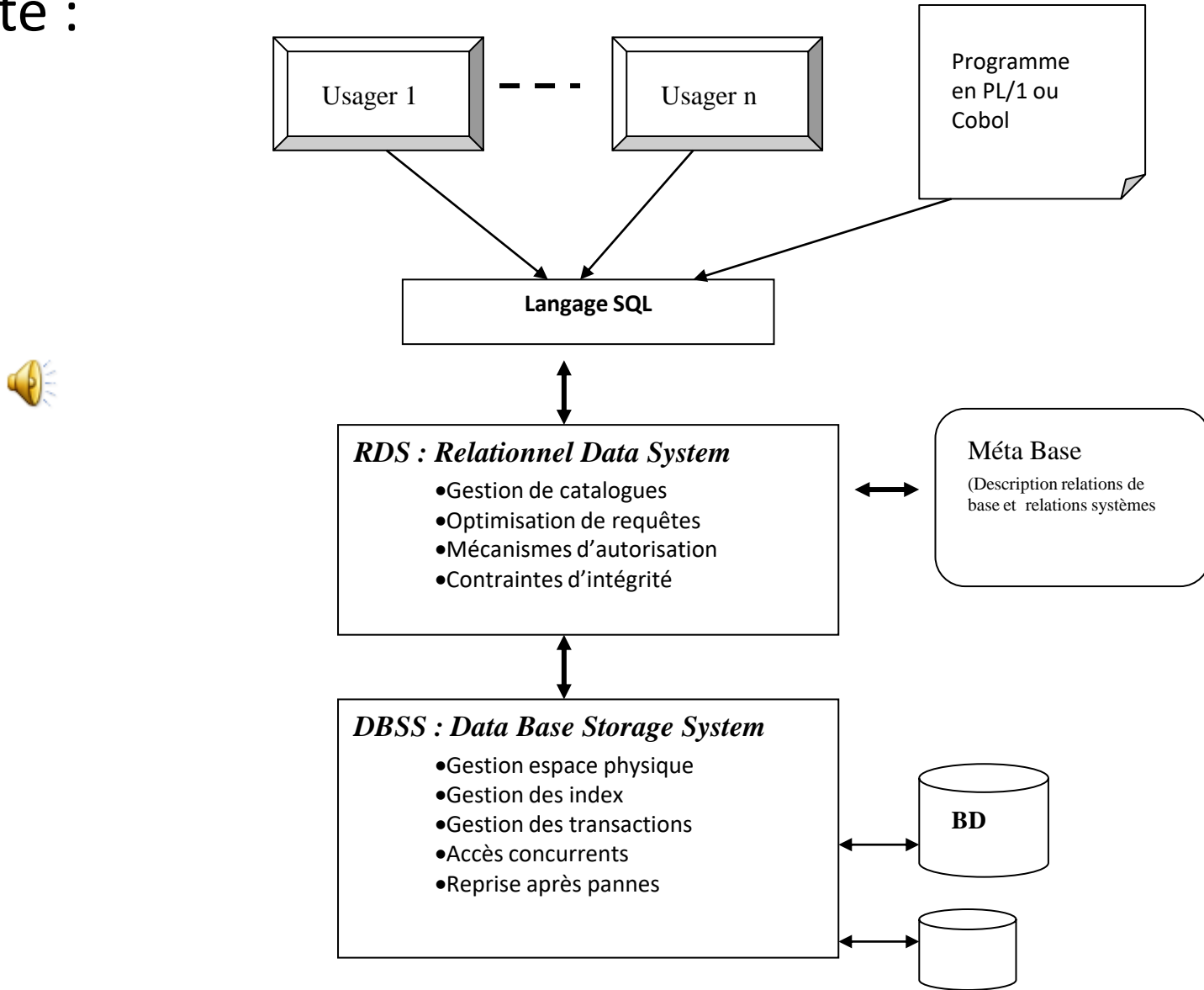
## 2.1. Architecture.

- L'architecture générale de « SYSTEM R » a la forme générale suivante :



## 2.1. Architecture.

- L'architecture générale de « SYSTEM R » a la forme générale suivante :





## 2.1. Architecture.

- L'architecture générale de « SYSTEM R » a la forme générale suivante :

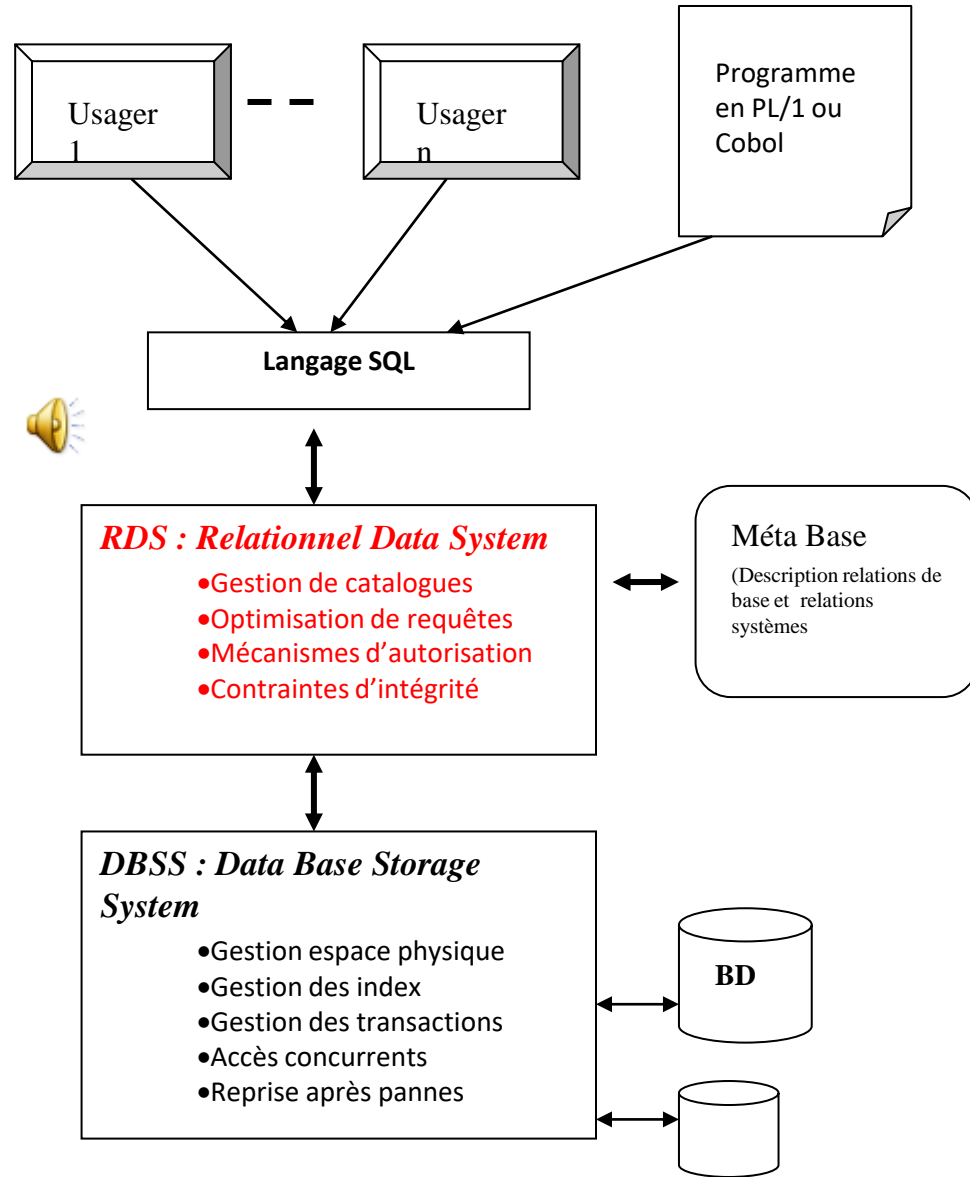
En réalité la structure de « System R » est relativement complexe, et le produit comporte un grand nombre de composants, que nous pouvons regrouper en deux grands sous-systèmes :



# 2.1. Architecture.

- L'architecture générale de « SYSTEM R » a la forme générale suivante :

le sous-système de données relationnel (RDS) qui gère globalement toutes les interactions avec les utilisateurs, et certaines fonctions systèmes comme l'optimisation des requêtes ou le contrôle des autorisations. C'est ce sous-système qui prépare à l'exécution de la requête. Nous verrons ci-après le déroulement d'une requête.

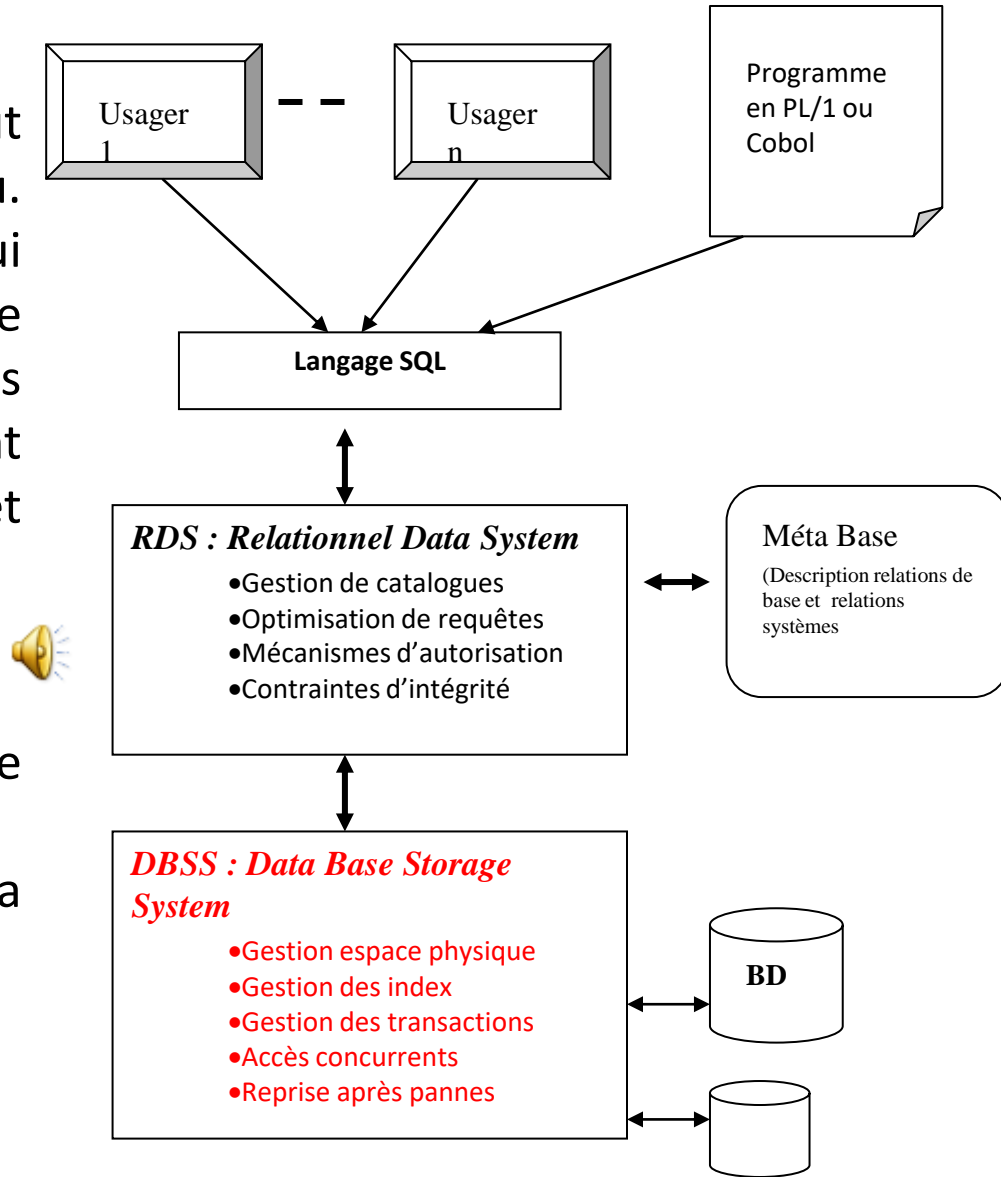


## 2.1. Architecture.

- L'architecture générale de « SYSTEM R » a la forme générale suivante :

Le **sous-système de stockage** qui peut être vu comme un S.G.B.D de bas niveau. Il comporte une interface interne qui fournit des opérateurs d'accès « un tuple à la fois » aux relations de base. Les appels aux fonctions du DBSS doivent spécifier explicitement les segments et index à utiliser. Le **DBSS gère** :

- Les allocations en mémoire secondaire
- La mémoire tampon
- Le contrôle de transactions (concurrence et reprise après pannes)
- Le maintien automatique des index (la méthode d'accès supportée est VSAM).




## 2.2. Exécution d'une requête

De manière générale, une requête est analysée en suivant les étapes suivantes :

- Analyse syntaxique de la question
- Optimisation
- Génération d'un plan d'exécution de la requête



## 2.2. Exécution d'une requête

- Il existe deux manières d'interroger une base de données :
  - En Mode interactif.
  - En Mode programmé.

## 2.2. Exécution d'une requête

- Il existe deux manières d'interroger une base de données :



### a. En Mode interactif.

l'utilisateur **interroge directement** la base de données par des **requêtes en langage SQL**, directement interprétables.

## 2.2. Exécution d'une requête

- Il existe deux manières d'interroger une base de données :

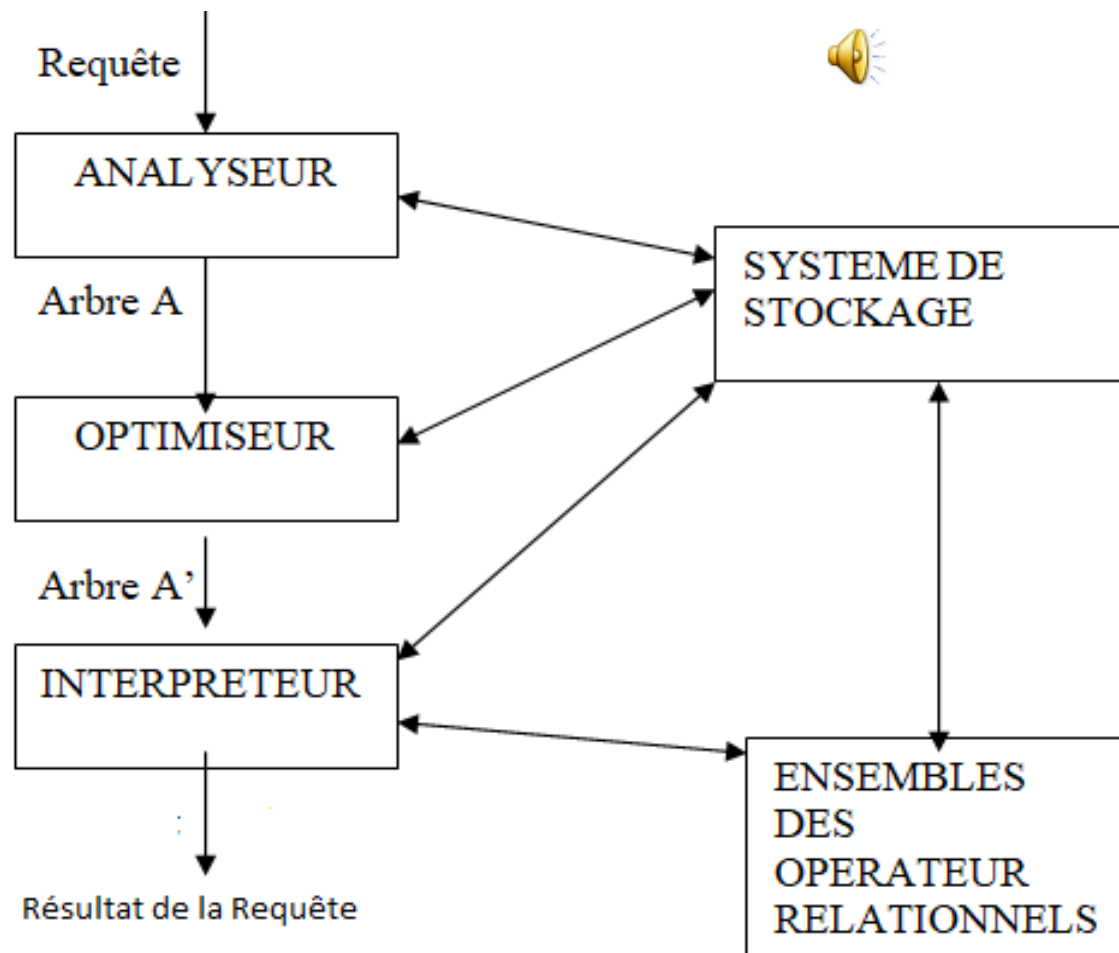
### a. En Mode interactif.



l'utilisateur interroge directement la base de données par des requêtes en langage SQL, directement interprétables. Un plan d'exécution de la requête est généré en passant **d'abord** par les phases **d'analyse syntaxique** et **d'optimisation** suivant le schéma ci-dessous :

## 2.2. Exécution d'une requête

### a. En Mode interactif.

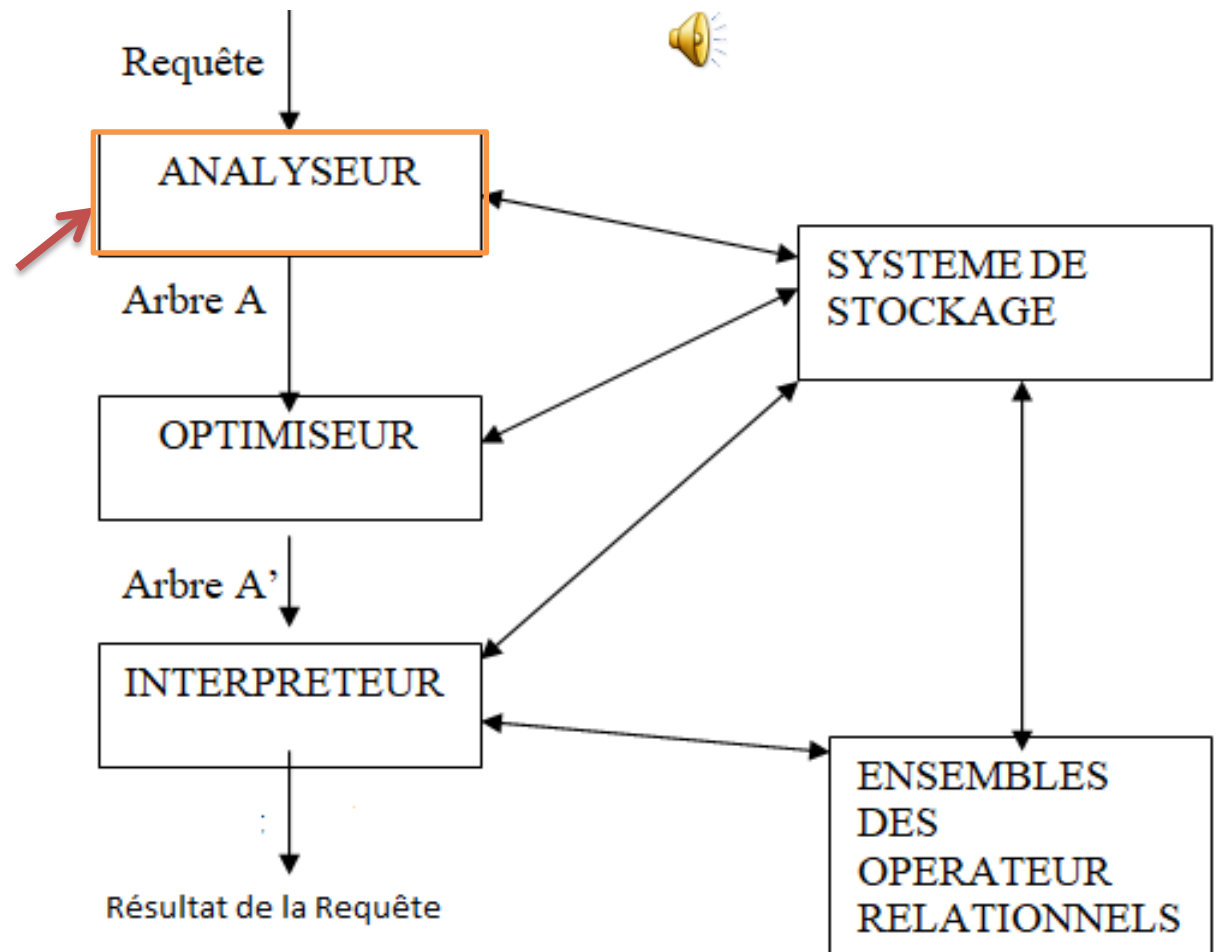




## 2.2. Exécution d'une requête

### a. En Mode interactif.

L'analyseur transforme la question sous forme d'arbre où **les nœuds** correspondent aux **opérateurs** et les **feuilles** aux **attributs** et aux **relations**.



## 2.2. Exécution d'une requête

### A . En Mode interactif.

**L'analyseur** transforme la question sous forme **d'arbre** où les nœuds correspondent aux opérateurs et les feuilles aux attributs et aux relations.



#### **Exemple :**

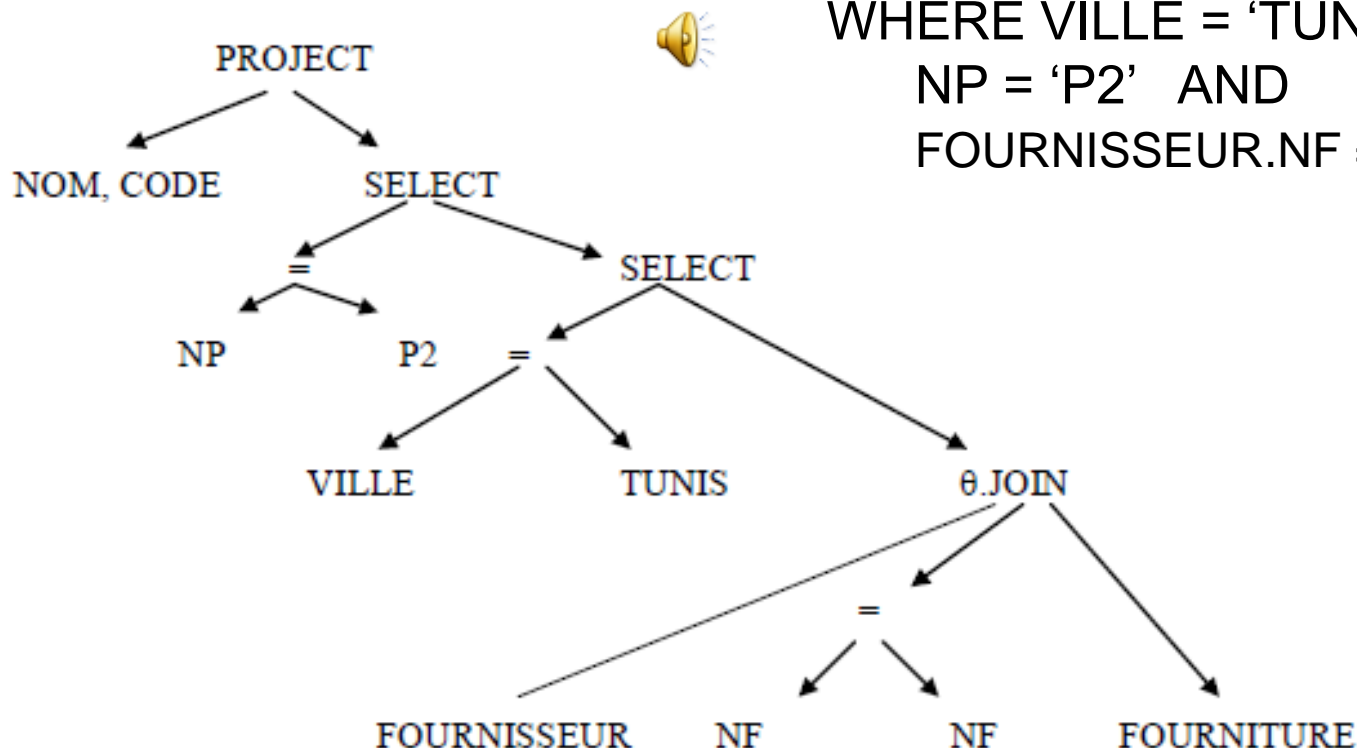
```
SELECT NOM, CODE  
FROM   FOURNISSEUR, FOURNITURE  
WHERE  VILLE = 'TUNIS' AND NP = 'P2' AND  
       FOURNISSEUR.NF = FOURNITURE.NF ;
```

## 2.2. Exécution d'une requête

### A . En Mode interactif.

#### Exemple :

SELECT NOM, CODE  
FROM FOURNISSEUR, FOURNITURE  
WHERE VILLE = 'TUNIS' AND  
NP = 'P2' AND  
FOURNISSEUR.NF = FOURNITURE.NF ;



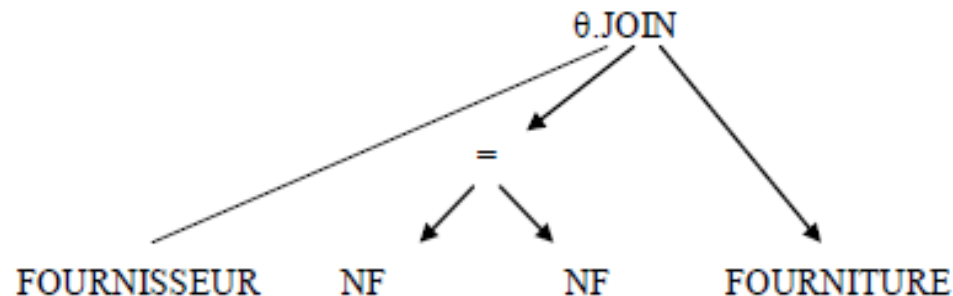
## 2.2. Exécution d'une requête

### A . En Mode interactif.

#### Exemple :



```
SELECT NOM, CODE  
FROM FOURNISSEUR, FOURNITURE  
WHERE VILLE = 'TUNIS' AND  
      NP = 'P2' AND  
      FOURNISSEUR.NF = FOURNITURE.NF ;
```



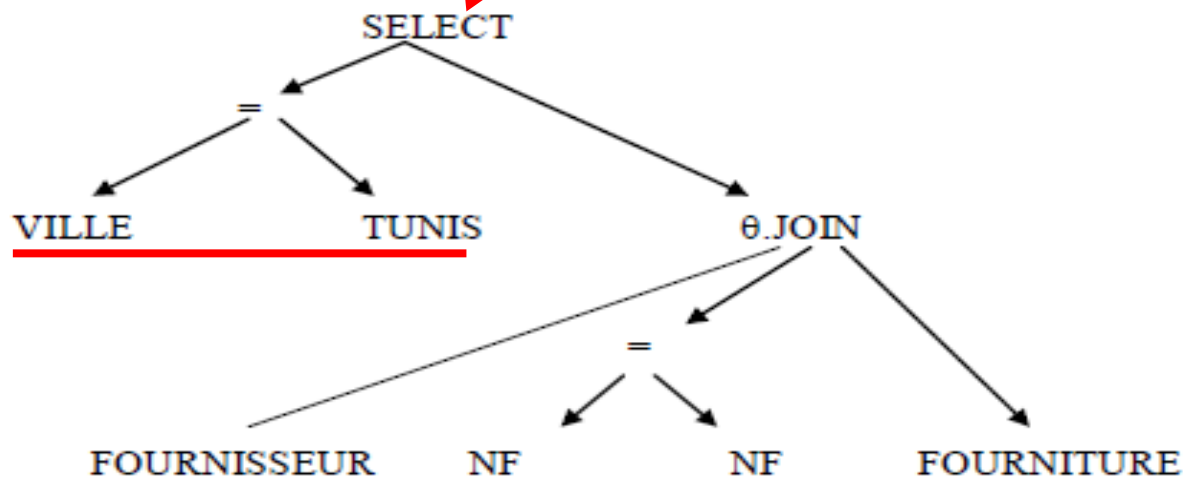
## 2.2. Exécution d'une requête

### A . En Mode interactif.

#### Exemple :



SELECT NOM, CODE  
FROM FOURNISSEUR, FOURNITURE  
WHERE **VILLE** = 'TUNIS' AND  
NP = 'P2' AND  
FOURNISSEUR.NF = FOURNITURE.NF ;



## 2.2. Exécution d'une requête

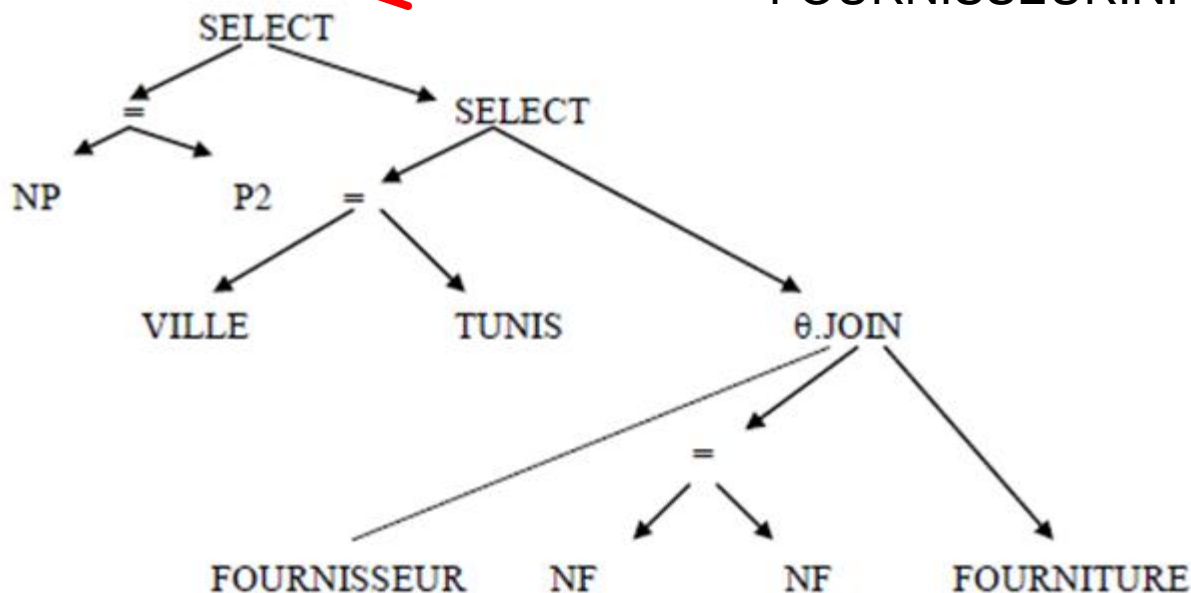
### A . En Mode interactif.

#### Exemple :



SELECT NOM, CODE  
FROM FOURNISSEUR, FOURNITURE  
WHERE VILLE = 'TUNIS' AND

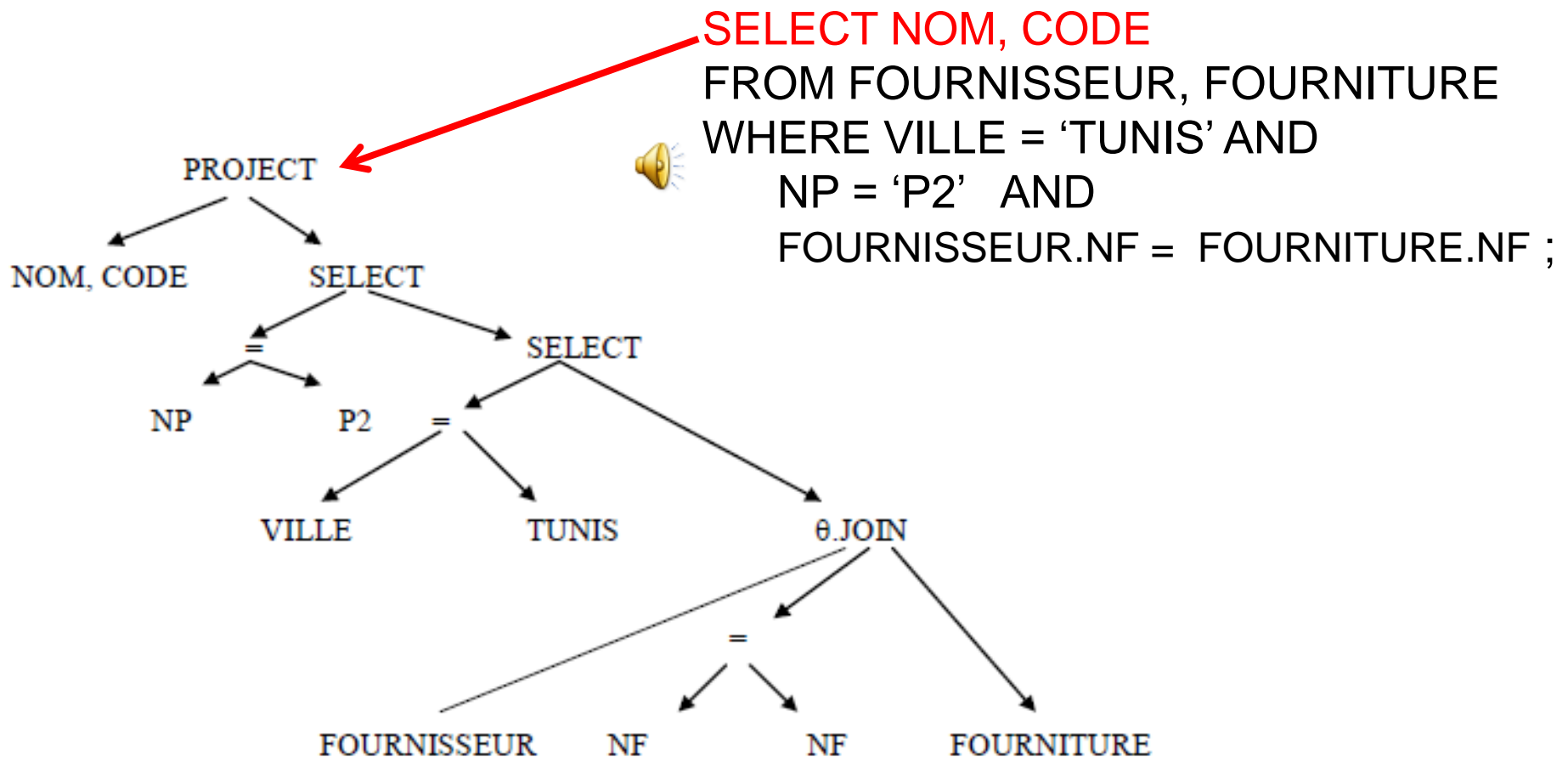
NP = 'P2' AND  
FOURNISSEUR.NF = FOURNITURE.NF ;



## 2.2. Exécution d'une requête

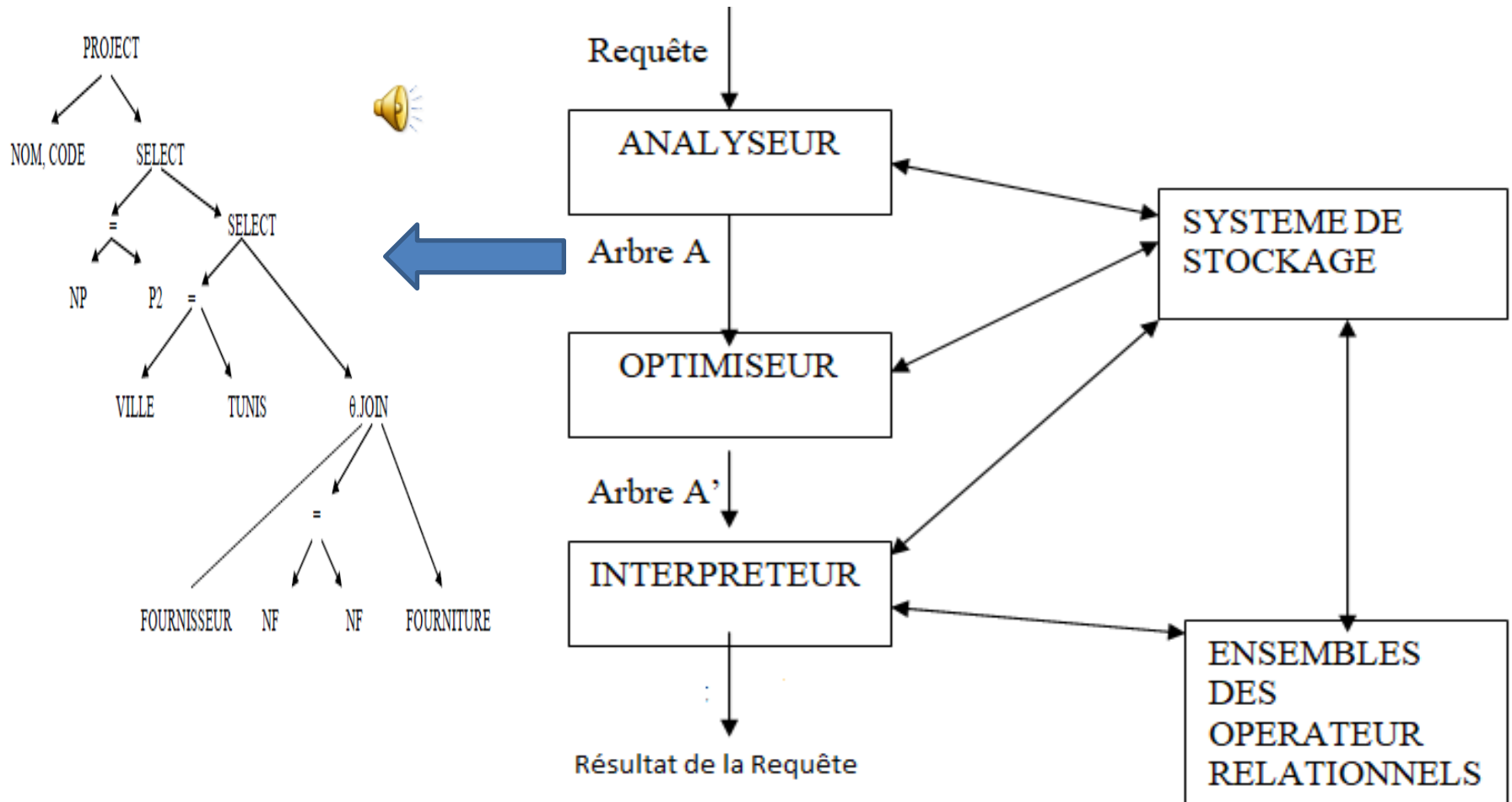
### A . En Mode interactif.

#### Exemple :



## 2.2. Exécution d'une requête

### A . En Mode interactif.





## 2.2. Exécution d'une requête

### A . En Mode interactif.



Différentes interprétation selon différents ordres d'application de ces opérateurs conduisent à des temps d'exploitation différents. D'où la nécessité de trouver des techniques qui permettent de trouver le meilleur ordre d'application des opérateurs, les meilleurs chemins d'accès aux relations. Ces techniques qui améliorent les performances des systèmes se retrouvent au niveau de l'optimiseur.

## 2.2. Exécution d'une requête

### A . En Mode interactif.

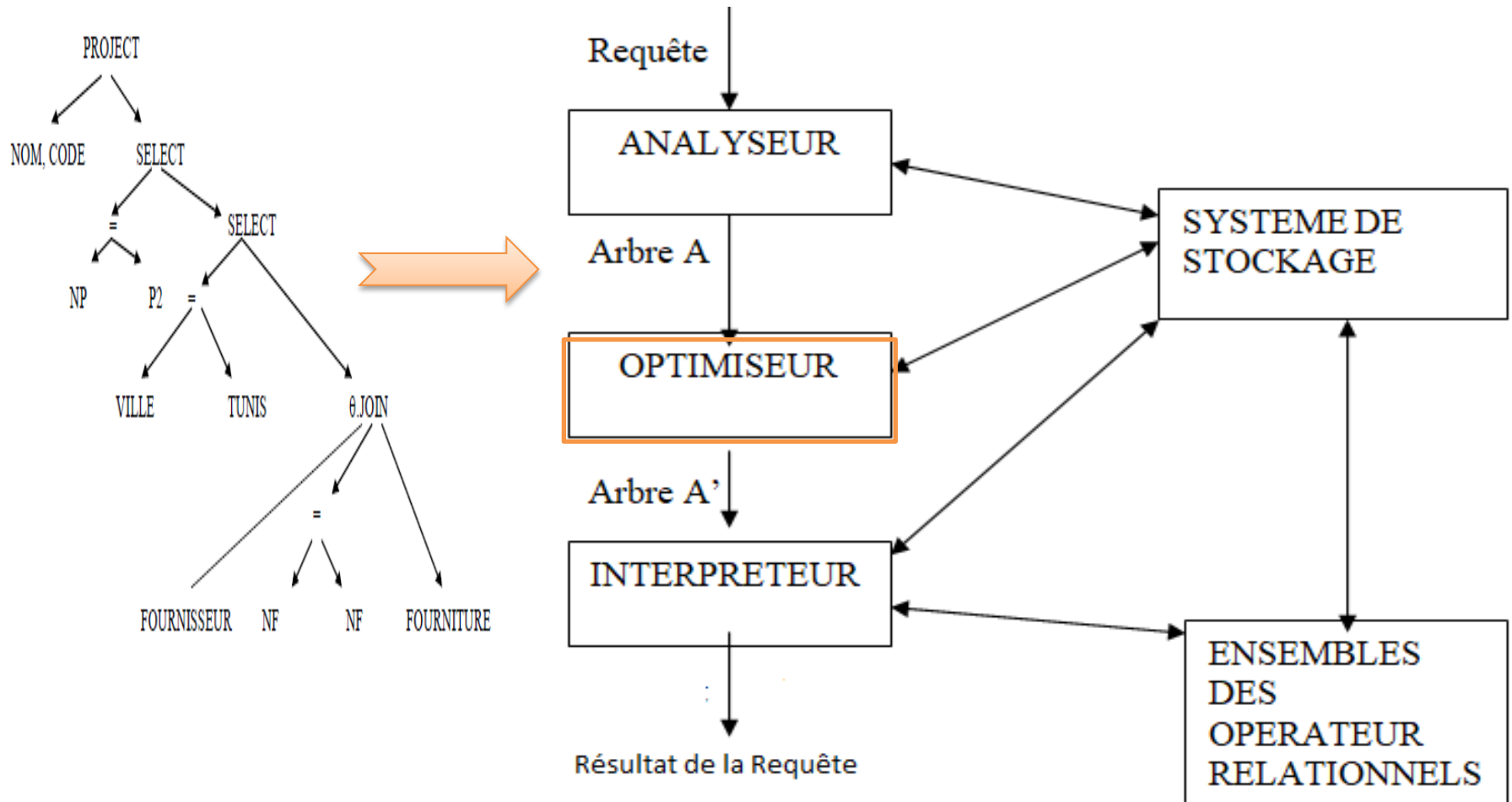


Différentes interprétation selon différents ordres d'application de ces opérateurs conduisent à des temps d'exploitation différents. D'où la nécessité de trouver des techniques qui permettent de trouver le meilleur ordre d'application des opérateurs, les meilleurs chemins d'accès aux relations. Ces techniques qui améliorent les performances des systèmes se retrouvent au niveau de l'optimiseur.

En sortie de **l'optimiseur** on aura donc un arbre optimisé. L'arbre précédent sera transformé par l'optimiseur comme suit :

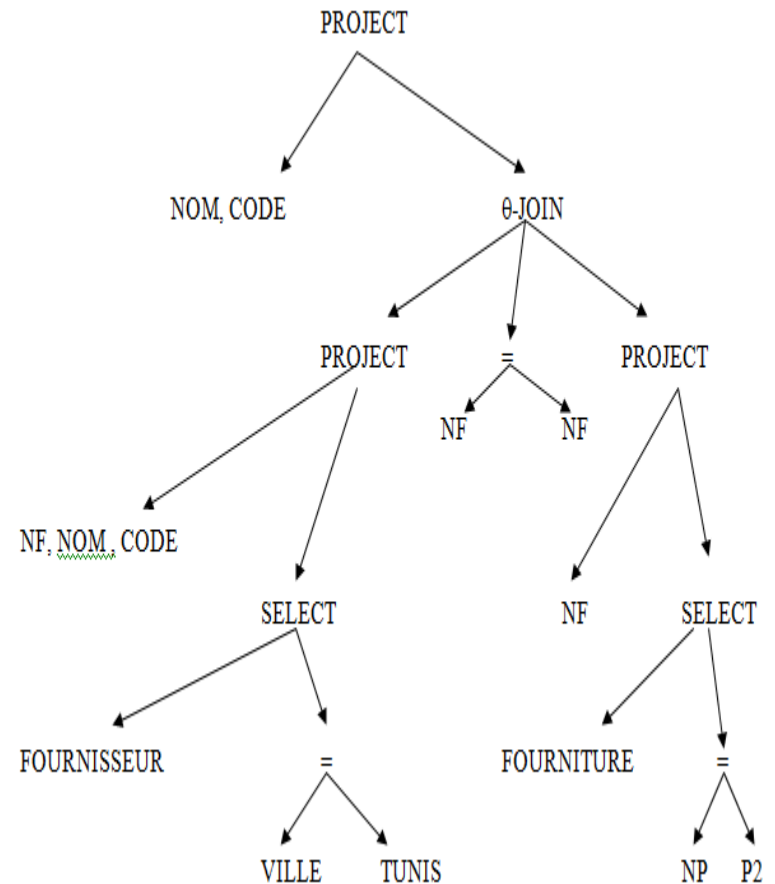
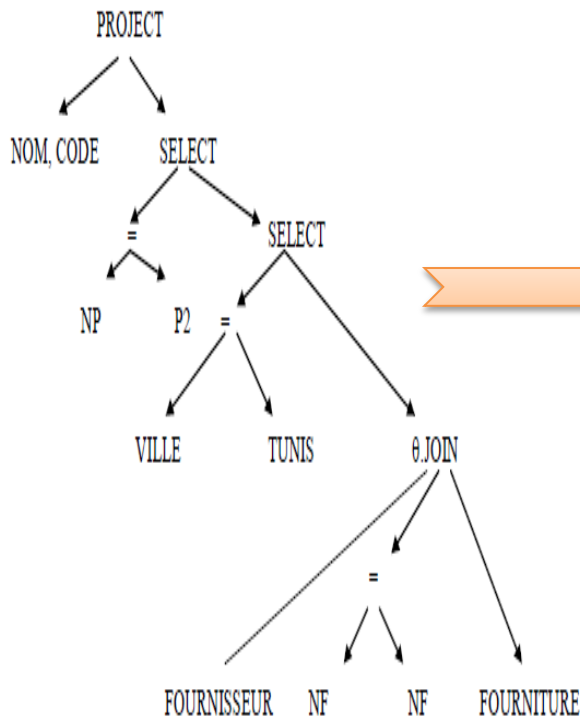
## 2.2. Exécution d'une requête

### A . En Mode interactif.



## 2.2. Exécution d'une requête

### A . En Mode interactif.

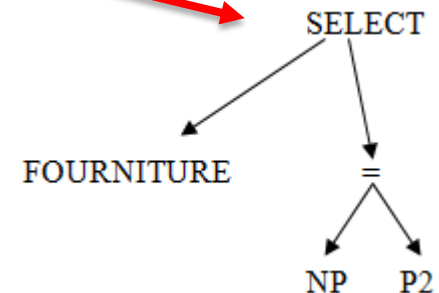
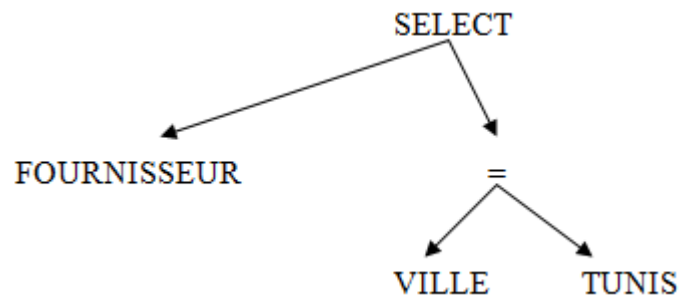


## 2.2. Exécution d'une requête

### A . En Mode interactif.

#### Exemple :

SELECT NOM, CODE  
FROM FOURNISSEUR, FOURNITURE  
WHERE VILLE = 'TUNIS' AND  
NP = 'P2' AND  
FOURNISSEUR.NF = FOURNITURE.NF ;



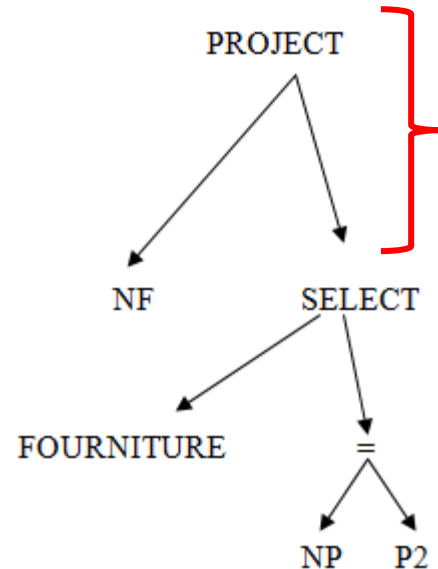
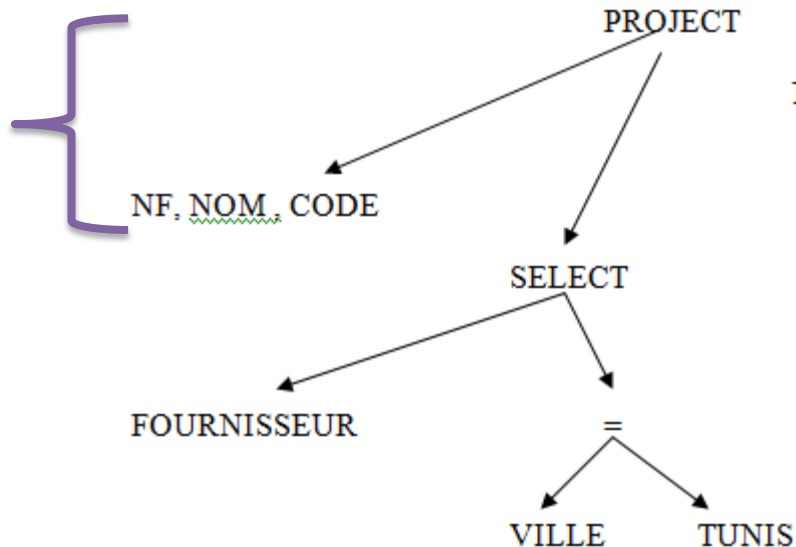
## 2.2. Exécution d'une requête

### A . En Mode interactif.

#### Exemple :



SELECT NOM, CODE  
FROM FOURNISSEUR, FOURNITURE  
WHERE VILLE = 'TUNIS' AND  
NP = 'P2' AND  
FOURNISSEUR.NF = FOURNITURE.NF ;



## 2.2. Exécution d'une requête

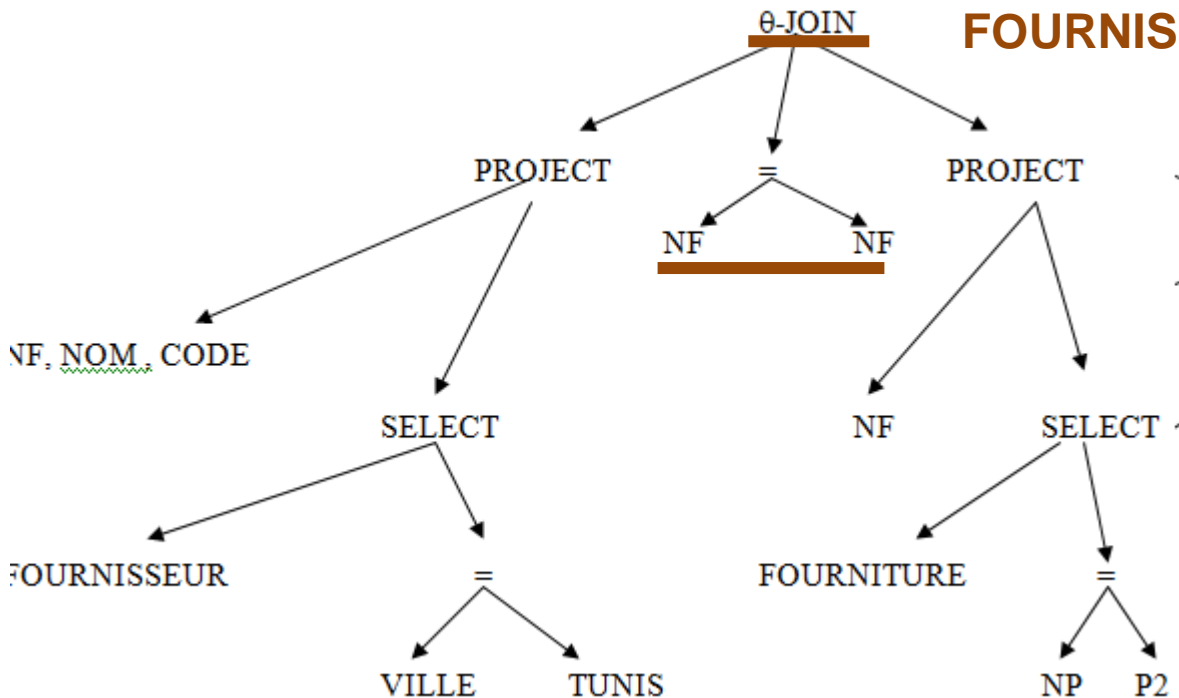
### A . En Mode interactif.

**Exemple :**



SELECT NOM, CODE  
FROM FOURNISSEUR, FOURNITURE  
WHERE VILLE = 'TUNIS' AND  
NP = 'P2' AND

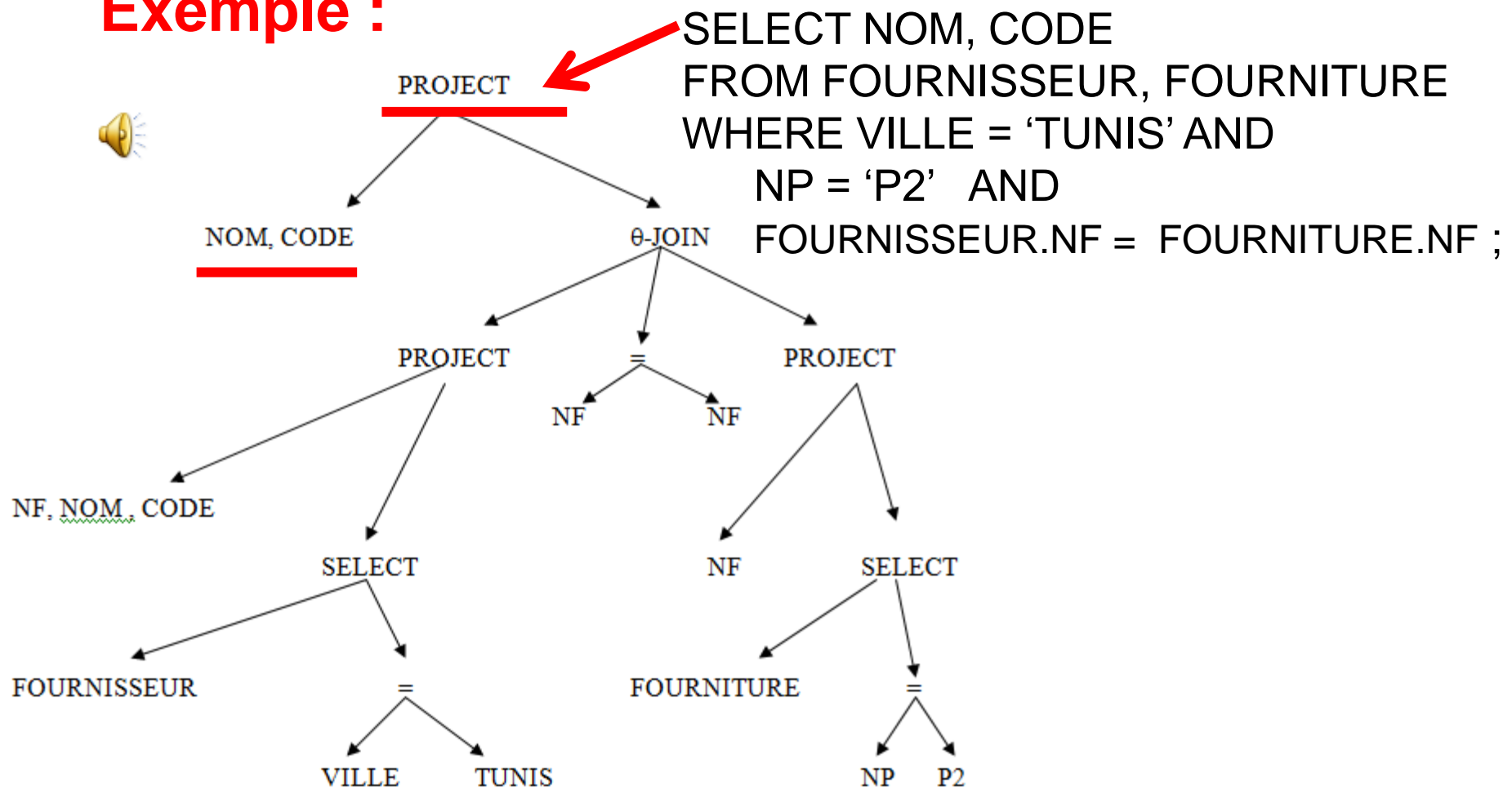
**FOURNISSEUR.NF = FOURNITURE.NF ;**



## 2.2. Exécution d'une requête

### A . En Mode interactif.

**Exemple :**

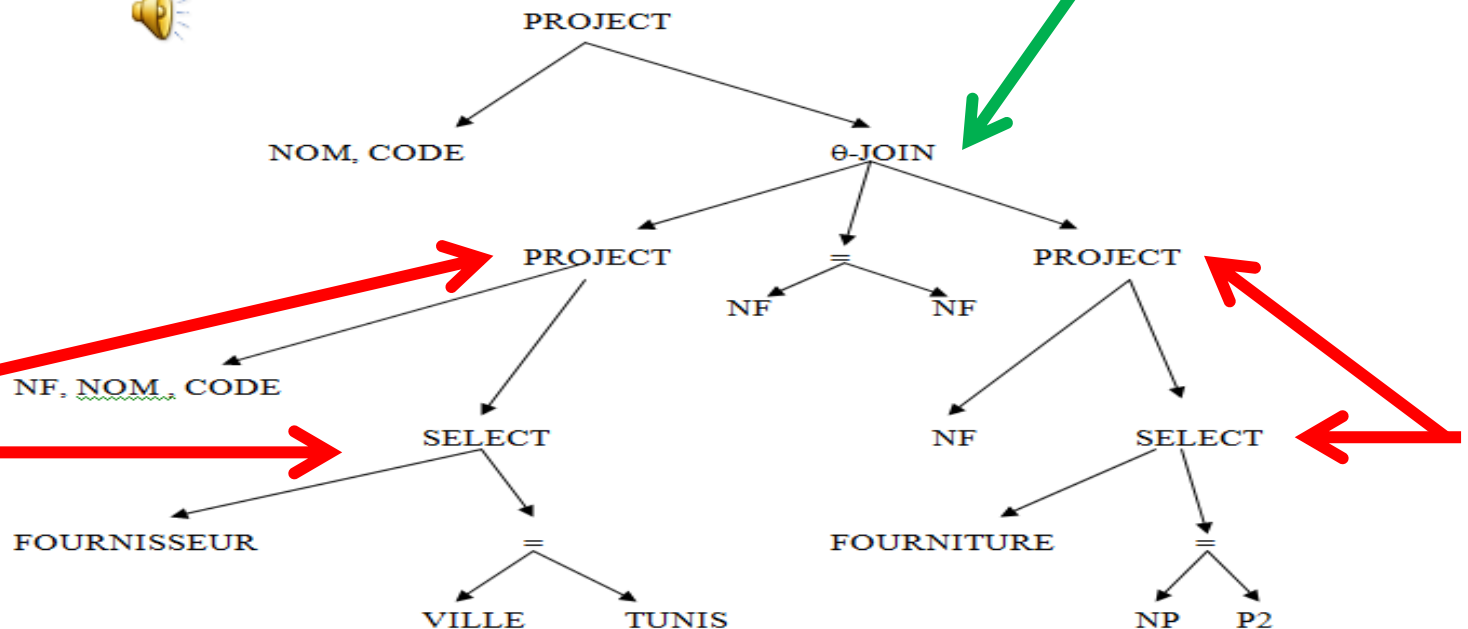




## 2.2. Exécution d'une requête

### A . En Mode interactif.

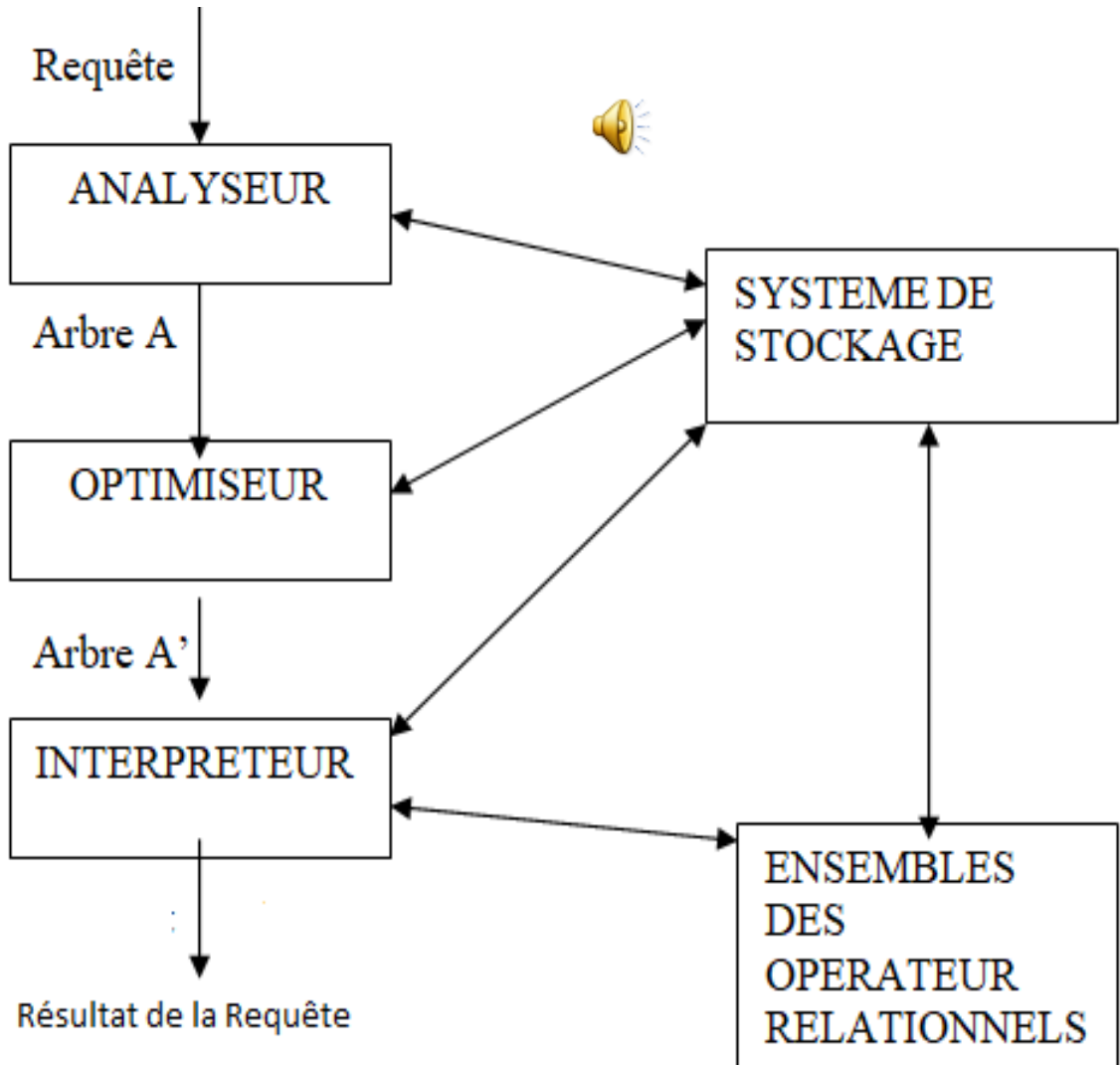
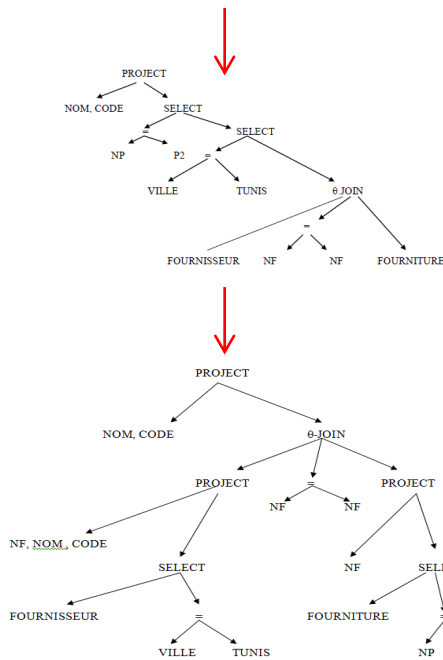
Cet arbre est obtenu par application des opérateurs de **réduction d'abord** en laissant les opérateurs de **jointure** à la fin si possible.



# 2.2. Exécution d'une requête

## A . En Mode interactif.

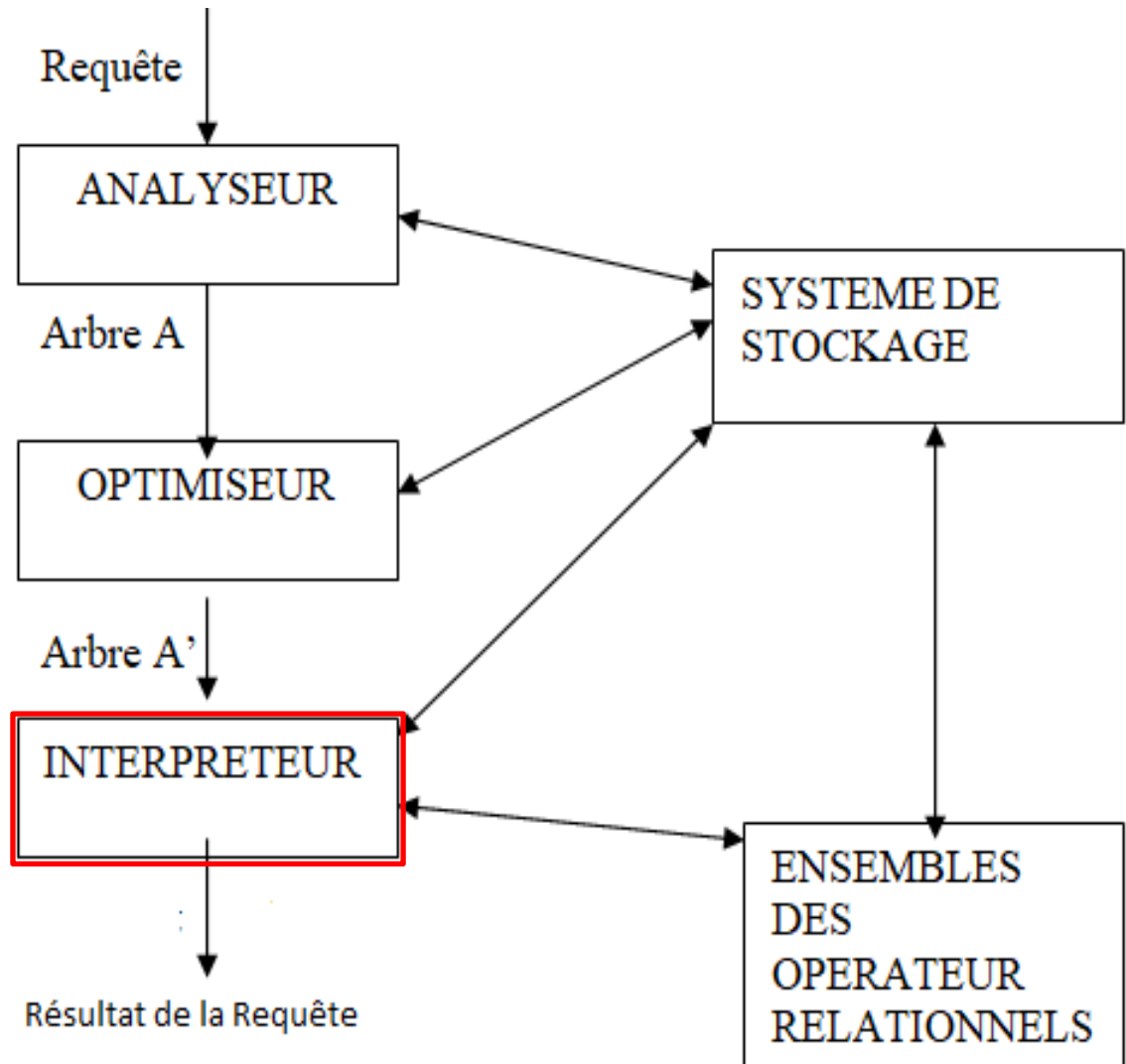
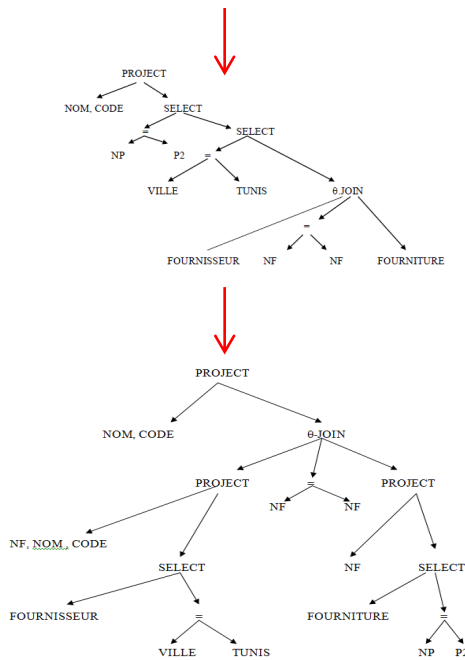
```
SELECT NOM, CODE  
FROM FOURNISSEUR, FOURNITURE  
WHERE VILLE = 'TUNIS' AND  
NP = 'P2' AND  
FOURNISSEUR.NF = FOURNITURE.NF ;
```



# 2.2. Exécution d'une requête

## A . En Mode interactif.

```
SELECT NOM, CODE
FROM FOURNISSEUR, FOURNITURE
WHERE VILLE = 'TUNIS' AND
      NP = 'P2' AND
      FOURNISSEUR.NF = FOURNITURE.NF ;
```

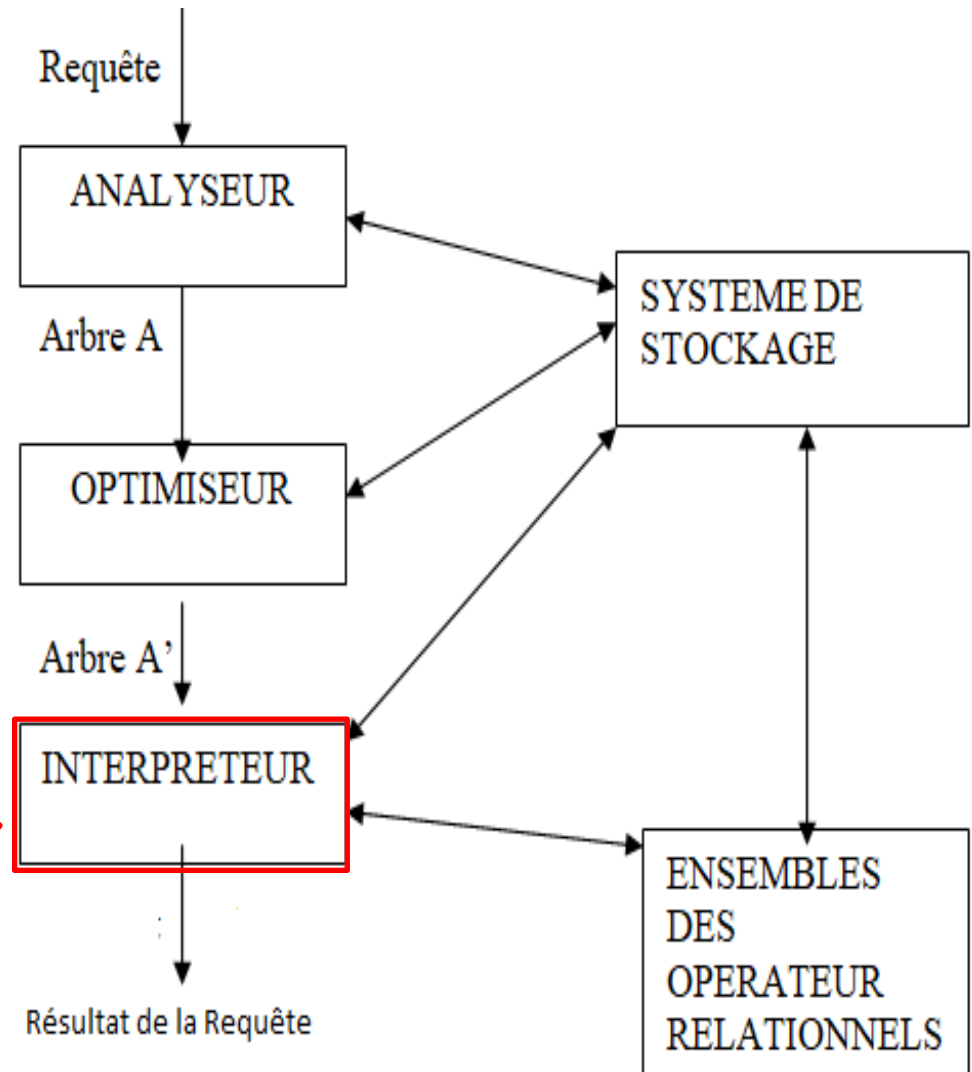
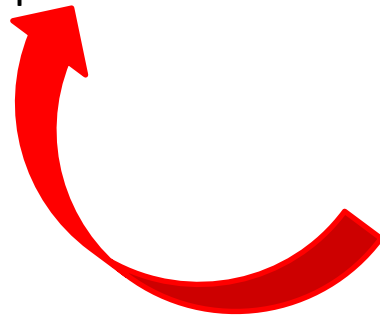


## 2.2. Exécution d'une requête

### A . En Mode interactif.



L'interpréteur de requêtes fournit le résultat de la requête en exécutant les opérateurs contenus dans l'arborescence à l'aide de procédure associée à ces opérateurs.



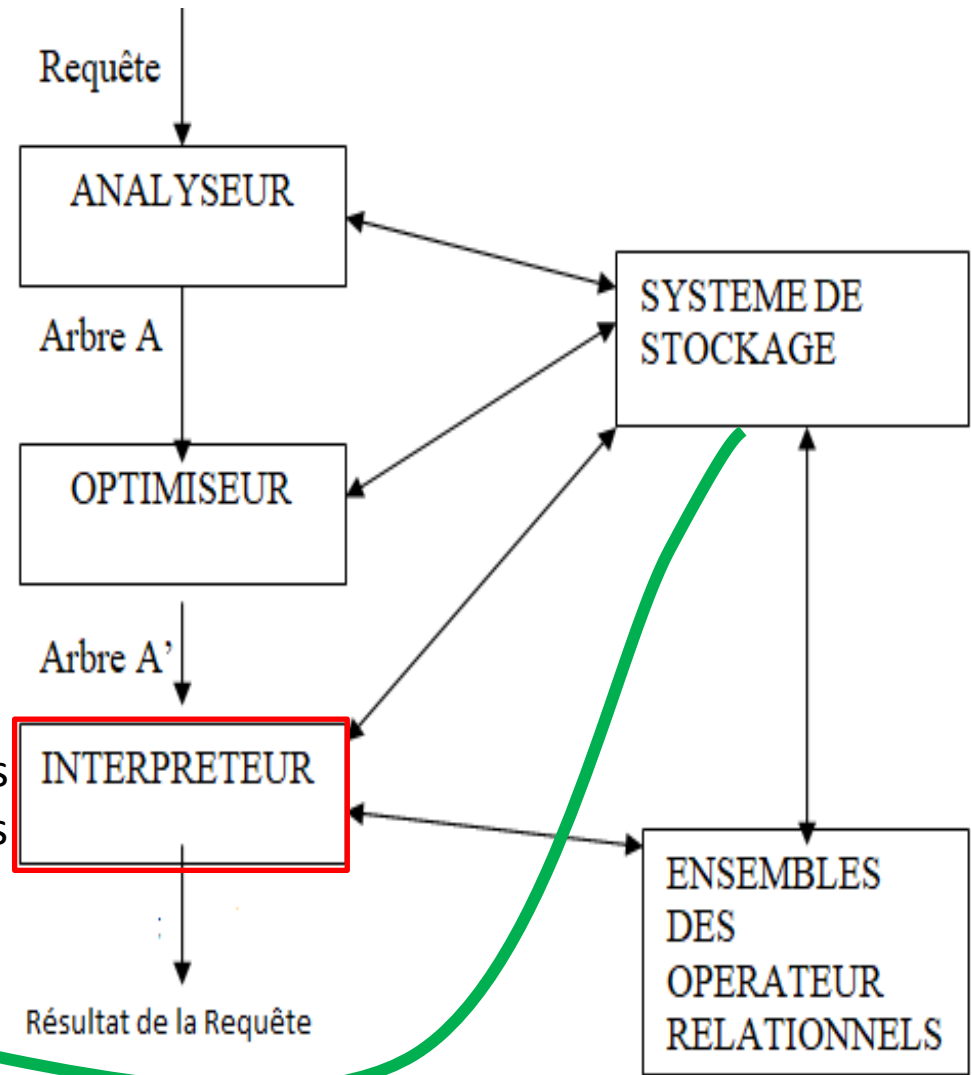
## 2.2. Exécution d'une requête

### A . En Mode interactif.



L'**interpréteur** de requêtes fournit le résultat de la requête en exécutant les opérateurs contenus dans l'arborescence à l'aide de procédure associée à ces opérateurs.

Le **système de stockage** doit permettre une organisation efficace des données et un accès rapide grâce à l'implémentation de méthodes d'accès appropriées



## 2.2. Exécution d'une requête

- Il existe deux manières d'interroger une base de données :
  - A. En Mode interactif.
  - B. **En Mode programmé.**



## 2.2. Exécution d'une requête

### B. En Mode programmé.

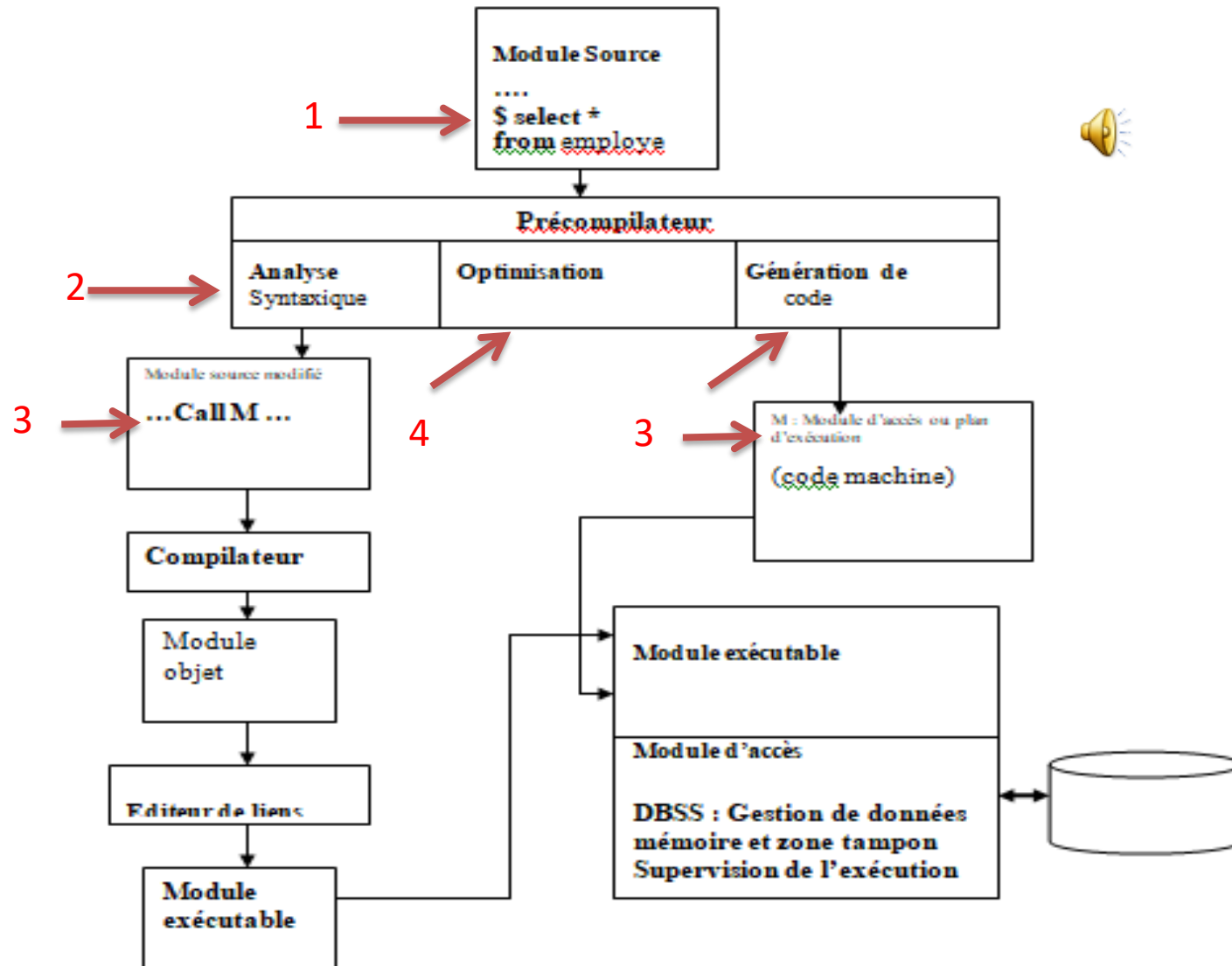
Les ordres SQL sont incorporés à un programme d'application écrit en langage de programmation PL/1, C, COBOL etc. appelés langages hôtes.

Dans ce cas la démarche nécessite une approche de précompilation. 

Le **précompilateur** est un processeur qui regroupe les instructions SQL du programme d'application dans un **module de requêtes de la BD**, et les remplace dans le programme initial par des appels (CALLs) du langage hôte à un superviseur d'exécution qui surveille les programmes SQL.

## 2.2. Exécution d'une requête

### B. En Mode programmé.





## 2.2. Exécution d'une requête

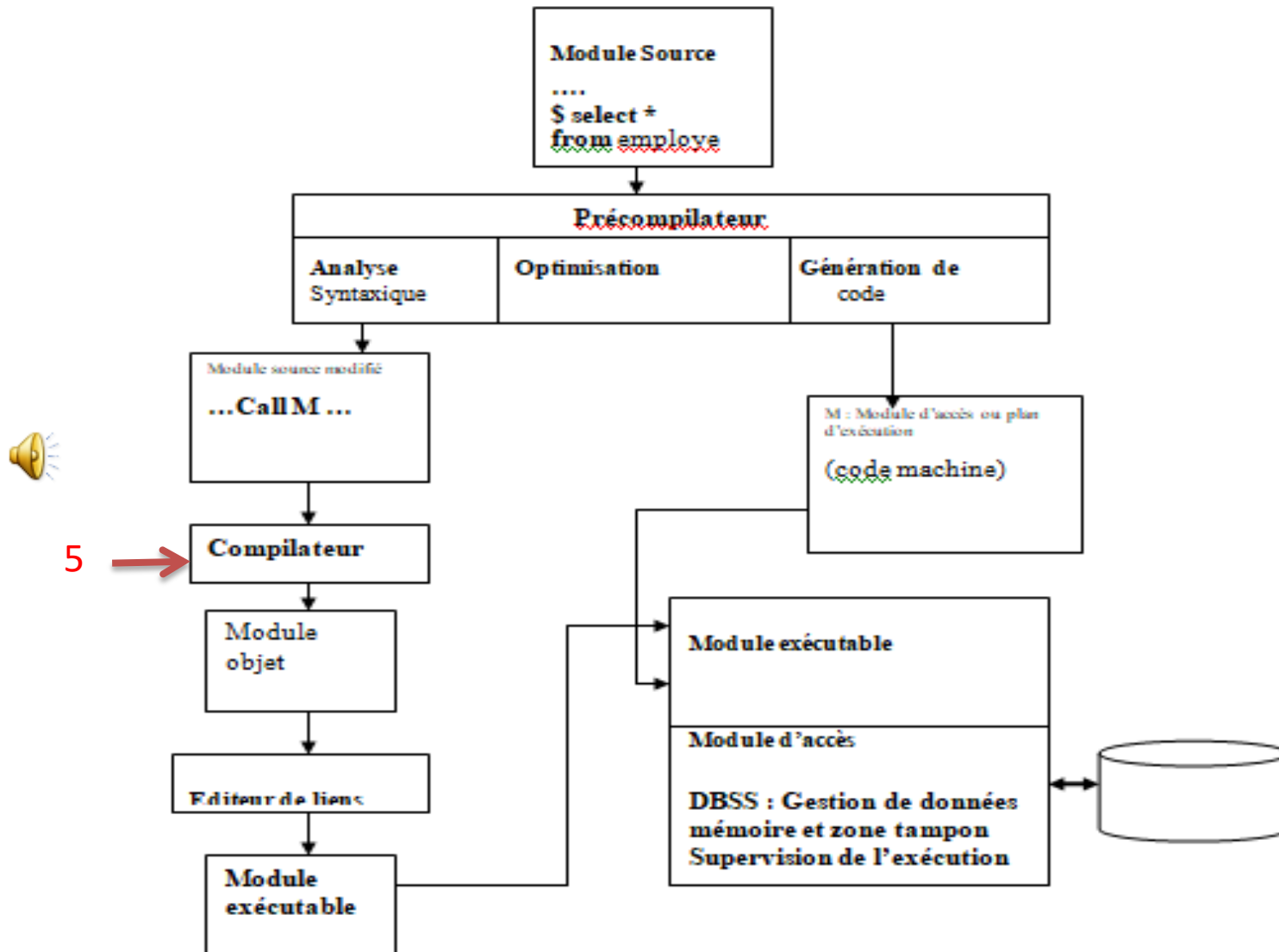
### B. En Mode programmé.



- **Fonctions du précompilateur :** Analyse le programme source, remplacement des ordres SQL par des instructions d'appels. Génération d'un module de requête de base de données qui passe par l'interpréteur SQL pour être exécuté.


## 2.2. Exécution d'une requête

### B. En Mode programmé.



## 2.2. Exécution d'une requête

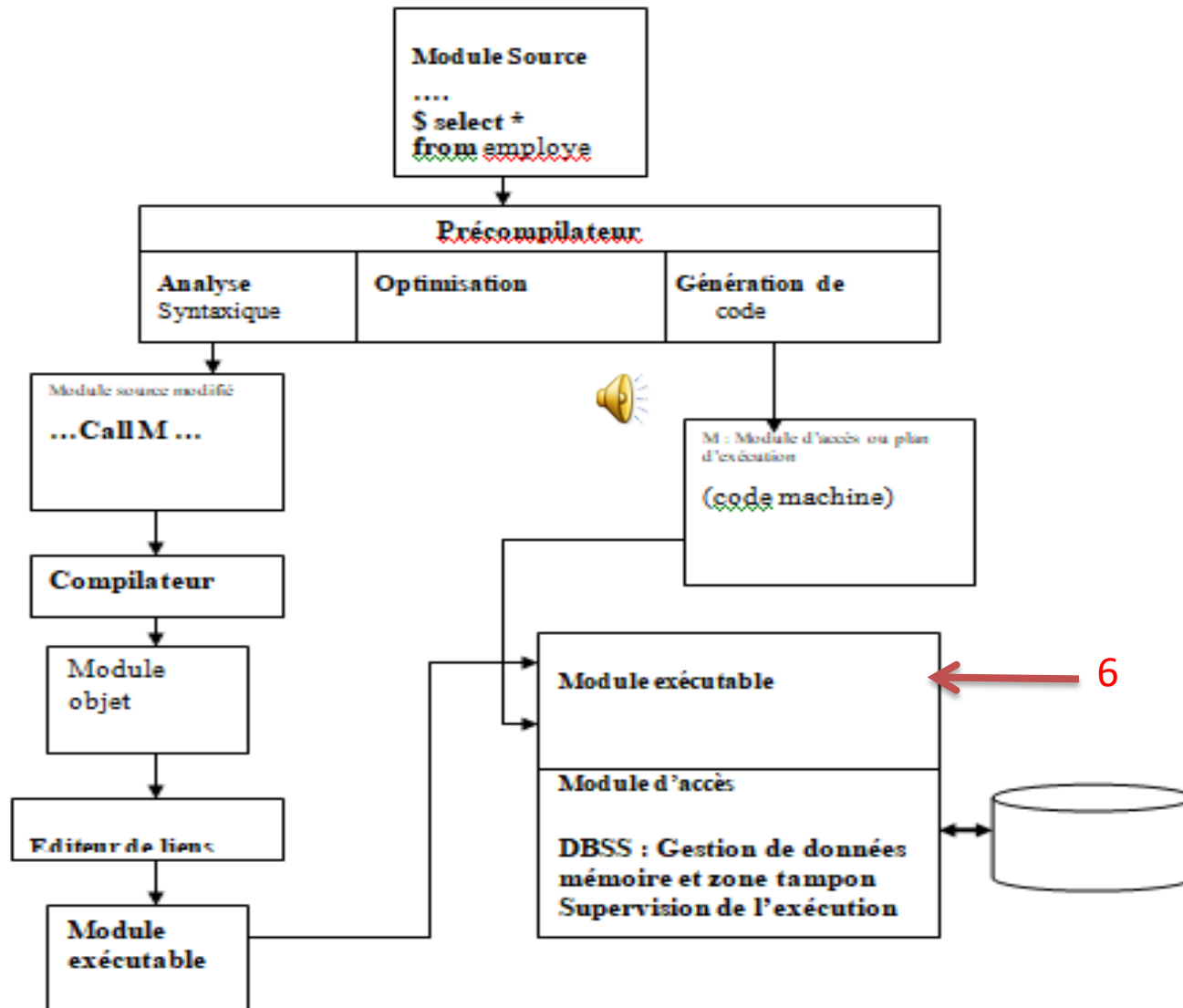
### B. En Mode programmé.

- **Fonctions du précompilateur :** Analyse le programme source, remplacement des ordres SQL par des instructions d'appels. Génération d'un module de requête de base de données qui passe par l'interpréteur SQL pour être exécuté. 

→ **Compilation du programme** avec édition de liens ayant pour résultat un module exécutable.


## 2.2. Exécution d'une requête

### B. En Mode programmé.



## 2.2. Exécution d'une requête

### B. En Mode programmé.

- **Fonctions du précompilateur** : Analyse le programme source, remplacement des ordres SQL par des instructions d'appels. Génération d'un module de requête de base de données qui passe par l'interpréteur SQL pour être exécuté.
- **Compilation du programme** avec édition de liens ayant pour résultat un module exécutable. 
- ➔ **L'exécution s'effectue par le sous-système de stockage** : le module exécutable s'exécute normalement, lorsque le premier call est atteint, le superviseur prend la main et exécute le module d'accès en invoquant le gestionnaire de données et le gestionnaire des tampons.

## Chapitre 2 :

# Architecture et Fonctions d'un Système Relationnel de Base de Données

