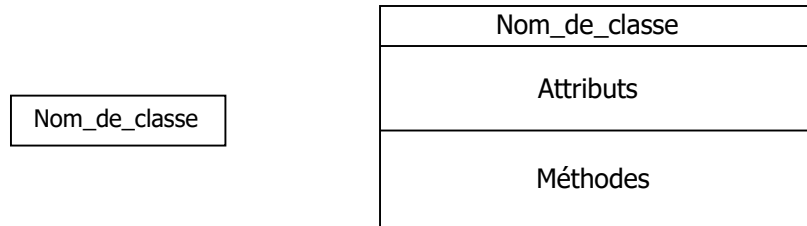


## Introduction :

**Les diagrammes de classes** représentent la structure statique d'un système informatique  
→ les classes, leurs structures internes et leurs relations.

Rappel : une classe est un type abstrait caractérisé par des propriétés (attributs et méthodes) communes à un ensemble d'objets et permettant de créer des objets ayant ces propriétés.

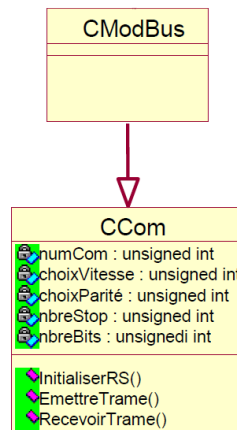
## Convention graphique :



## Relations entre classes :

Il y a 4 types de relations possibles.

## Héritage :



La classe CModBus hérite de la classe CCom

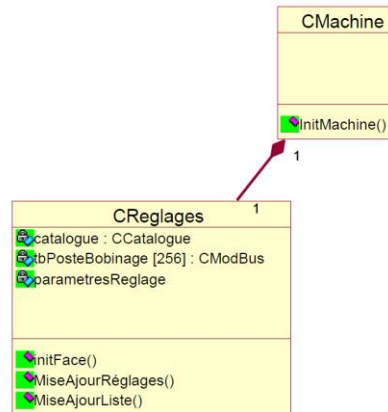
Déclaration en C++ : ***class CModBus : public CCom***  
***{***  
     ...  
***};***

Remarque : le terme exact en UML est généralisation au lieu d'héritage

**Définition agrégation :**

Il s'agit d'une relation entre deux classes, spécifiant que les objets d'une classe sont des composants de l'autre classe.

Une relation d'agrégation permet donc de définir des objets composés d'autres objets. L'agrégation permet d'assembler des objets de base, afin de construire des objets plus complexes.

**Composition (ou agrégation par valeur) :**

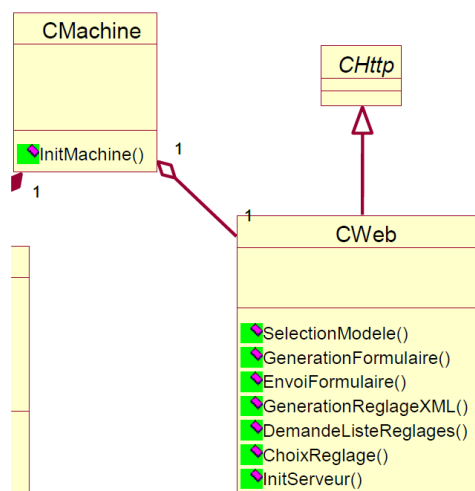
Un objet Cmachine utilise un objet de la classe CReglages.

Contrainte : la vie d'un objet CReglages est liée à celle de l'objet CMachine.

Déclaration en C++ :

```

class CMachine
{
private :
    CReglages reglage ;           // ou CReglages *reglage ;
    ...
};
  
```

**Agrégation par référence :**

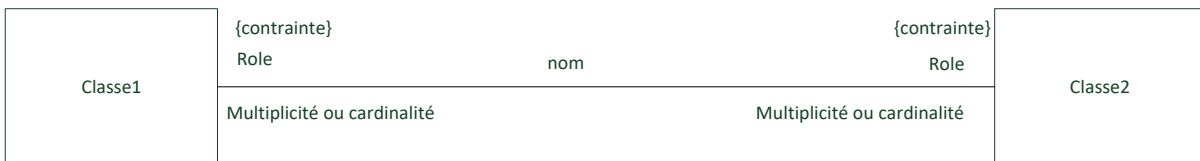
Un objet Cmachine utilise un objet de la classe CWeb.

Déclaration en C++ :

```
class CMachine
{
    CWeb &navigateur ;           // ou CWeb *navigateur;
    ...
};
```

### Association :

Une association entre classes indique qu'une classe doit « connaître » une autre classe pour pouvoir remplir sa tâche.



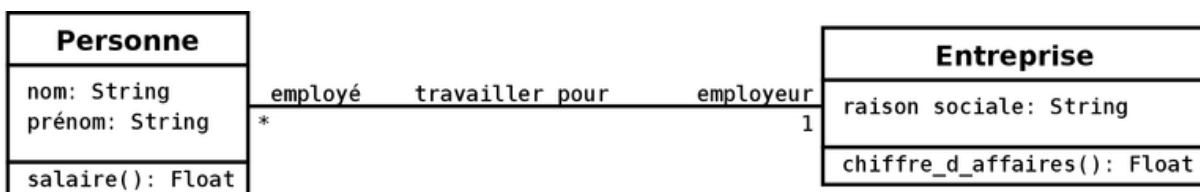
La représentation graphique est un trait plein ; elle **peut** posséder une flèche aux extrémités (ou à une seule extrémité) pour représenter le sens de navigation (unidirectionnelle ou bidirectionnelle).

Au milieu de l'association on peut avoir un nom qui correspond souvent à un verbe à l'infinitif.

Aux extrémités, on peut avoir les rôles (joués dans l'association) ou les contraintes (entre accolades).

Remarque : pour le rôle → nom de l'objet

Exemple :



La relation d'association en C++ sera réalisée par un pointeur ou une référence.

```
class Entreprise
{
    Personne &employé ;           // ou Personne *employé;
};
```

**Cardinalité ou multiplicité :**

Précise le nombre d'instances de classe (=objets) qui participent à une relation.

Exemple :

