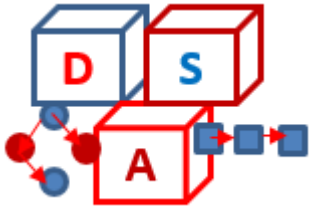


Structure de données

Les types structurés

Application gestion Ventes-V7

Version Février-2019

Abdelwahab Naji - **Enseignant chercheur**

Abdelwahab.naji@gmail.com

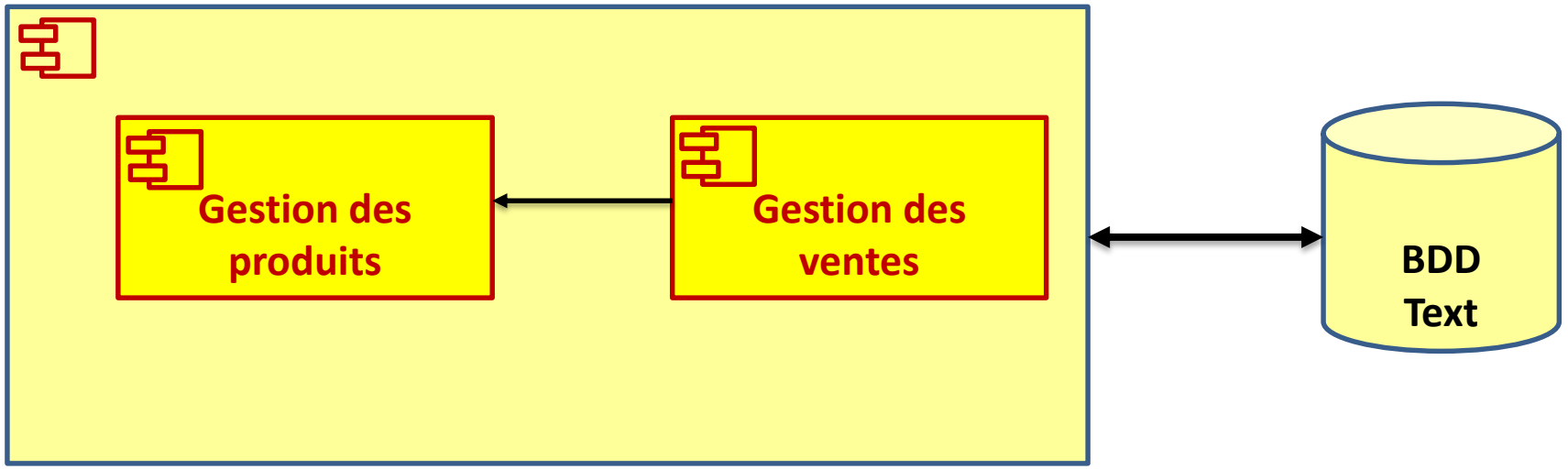
Laboratoire Signaux, Systèmes Distribués et Intelligence Artificielle (SSDIA)

ENSET de Mohammedia, Université Hassan II de Casablanca

https://www.youtube.com/c/AbdelwahabNaji

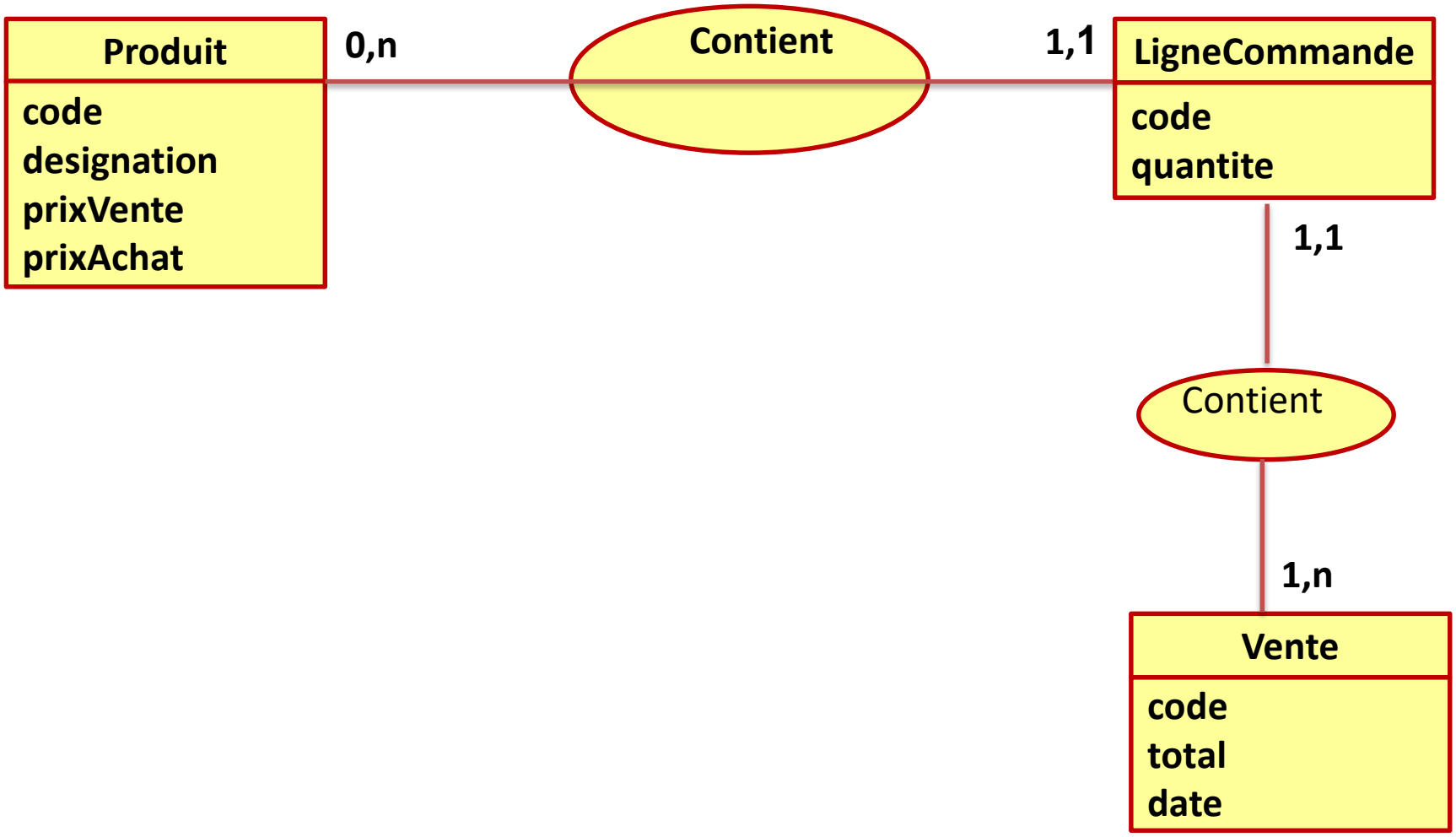
Gestion des ventes

- **Architecture de l'application**



1. Définir les données à traiter: modèle de données
2. Définir les fonctions: algorithmes

Gestion des ventes (Modèle 2)



Modèle de données (données à gérer) Merise

Rappel

```
Initialisation de donnees--------------

designation:    AAAA
designation:    BBBBB
designation:    CCCCCC
designation:    DDDD

--------LISTE DES PRODUITS------------

Code    Designation     Prix-Achat      Prix-Vente
20      AAAA            200.00          230.00
21      BBBBB           300.00          360.00
22      CCCCCC          400.00          490.00
23      DDDD            500.00          620.00
saisie de la vente-----------

code  quantite
Entrer ligne 1: 21

BBBBB    300.00  quantite:2
Entrer ligne 2: 23

DDDD     500.00  quantite:3
DÚtail de la vente-----------

code Designation quantite prix   s.total
21      BBBBB   2       360.00  720.00
23      DDDD    3       620.00  1860.00
-----------------Total:1540.00
```

- Les données et la mémoire
- Les pointeurs
- Les fonctions
- Les tableaux
- Les chaines de caractères

Problématique - exemple

```
saisirDonneesVente(int codeProduitLc[],int codeProduit[],char
*designation[],double prixVente[],int *quantite,int NPRODUITS,int
NLIGNES){
```

```
int main(){

const int NPRODUITS=4;
const int NLIGNES=2;
 int codeProduit[NPRODUITS];
 char* designationProduit[NPRODUITS];
 double prixAchatProduit[NPRODUITS];
 double prixVenteProduit[NPRODUITS];
int codeProduitLC[NLIGNES];
int quantite[NLIGNES];
double totalVente;
```

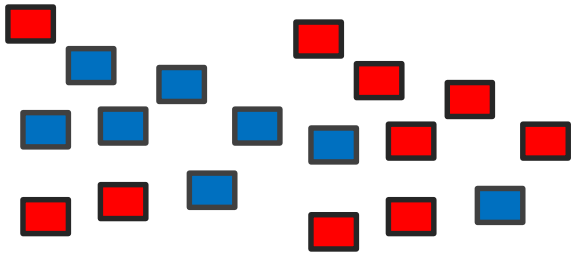
Problématique

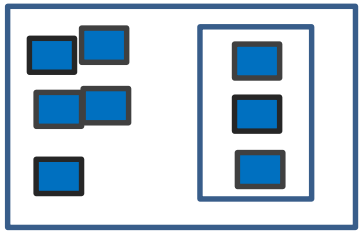
En se limitant à l'usage des types primitifs,

- **Beaucoup de variables indépendantes à utiliser ce qui rend difficile leurs gestion**
- **L'échange de données entre fonctions devient difficile à gérer: beaucoup de paramètres**
- **plus la taille du problème augmente, plus la lisibilité du code devient difficile, donc la maintenance devient un travail lourd à faire**

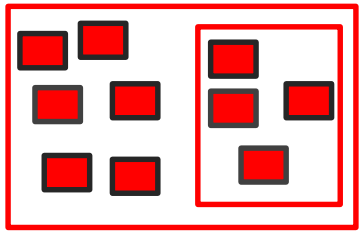
Les structures de données

- **Plusieurs données sont à regrouper dans la même entité appelée structure**
- **Une structure se compose d'un ensemble de données (de type primitif ou composé)**
- **Les structures données améliore la lisibilité du code et facilite son maintenance**
- **Le langage C réserve le mot clé struct pour créer de nouvelle structures**





Structure pour le Client



Structure pour le Produit

Données de type primitif pour représenter un Produit

Données de type primitif pour représenter un Client

Syntaxe en C d'une structure de données

- Syntaxe en C

```
struct Nom_structure{
 Type1  donnée1;
 Type2 donnée2;
…
};
```

```
struct produit{
  int codeProduit;
  char *designation;
  double prixAchat;
   double prixVente;
};
```

- Le mot clé struct est utilisé pour créer une nouvelle structure
- Nom_structure: représente le nom de la structure de données
- Type i: représente une donnée simple (type primitif) ou complexe (structure)

Syntaxe en C d'une structure de données

- Syntaxe en C

```
typedef struct Nom_structure{
 Type1  donnée1;
 Type2 donnée2;
…
}NomType;
```

```
typedef struct produit{
 int codeProduit;
 char *designation;
 double prixAchat;
  double prixVente;
}Produit;
```

Créer une variable d'une structure de données

1) Créer une variable sans Utiliser le mot clé typedef

```
struct produit{
  int codeProduit;
  char *designation;
  double prixAchat;
   double prixVente;
};
```

```
int main(){
struct produit p;
return 0;
}
```

2) Utiliser le mot clé typedef

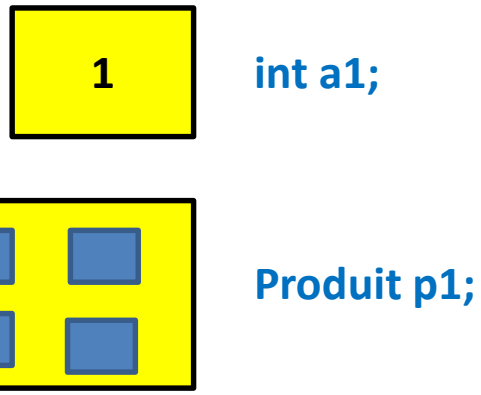
```
 typedef struct produit{
  int codeProduit;
  char *designation;
  double prixAchat;
   double prixVente;
}Produit;
```

```
int main(){
Produit p;
return 0;
}
```

Dans la suite du cours, nous allons utiliser cette deuxième méthode

Donnée d'une structure vs donnée de type primitif

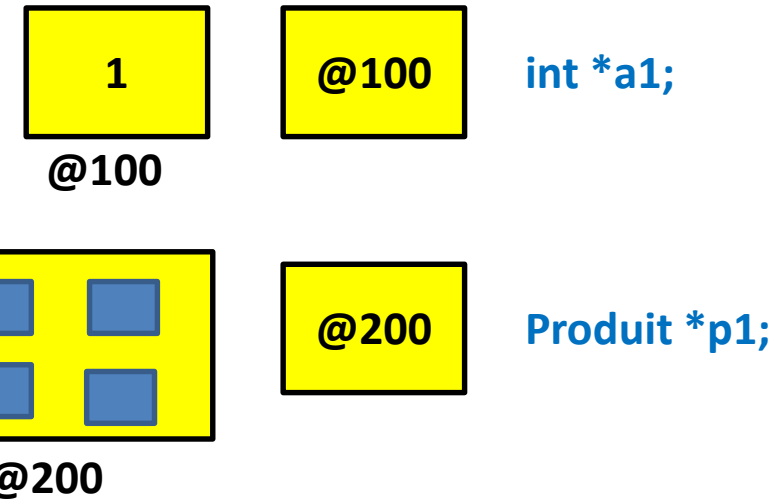
```
int main(){
 int a1;
Produit p1;

return 0;
}
```



- Donner le détail du p1 (adresse, taille, contenu)

Donnée d'une structure vs donnée de type primitif

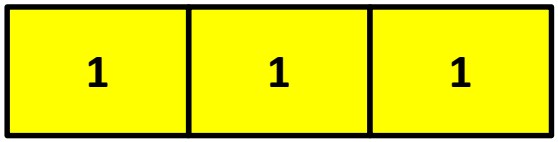
```
int main(){
int *a2;
Produit *p2;
return 0;
}
```



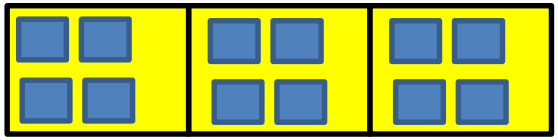
- Donner le détail du *p2 (adresse, taille, contenu)

Donnée d'une structure vs donnée de type primitif

```
int main(){
int Tab1[5];
Produit Tab2[5];
return 0;
}
```



int a1[3];



Produit p1[3];

Gestion des ventes – V7

En utilisant les structures et les tableaux, réaliser le module suivant:

- **Gestion des produits**
  - **Créer un produit**
  - **Afficher le détail d'un produit**
  - **Afficher la liste des produits**