

INTERFACES HOMME-MACHINE

Sommaire

1) *Problématique de la C.H.M*

2) Modèles d'IHM

3) Méthode de conception d'une IHM

4) Ergonomie des IHM

5) Maquettage, prototypage, évaluation

6) Interfaces adaptatives, futur des IHM, conclusion

INTERFACES HOMME-MACHINE

Chapitre 2 : Modèles d'IHM

2.1) Introduction

2.2) Modèle langage

2.3) Modèle Entrée / Sortie

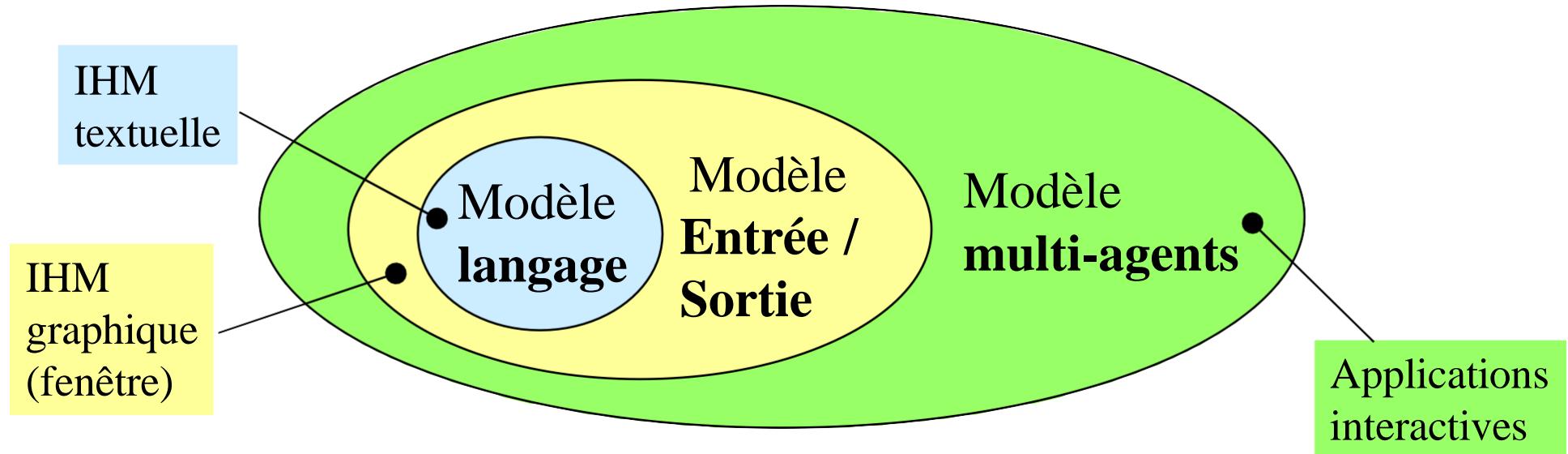
2.4) Modèle Multi-agents

2.1) Introduction

❑ un modèle ? pourquoi? pour qui ?

- ❖ pour le **concepteur** de l'IHM
- ❖ objet : fournir une **structure générique** d'IHM qui décrit
 - les **flux** de données
 - les étapes de **transformations** des données
 - l'**agencement** des composants qui les réalisent

❑ 3 modèles complémentaires



INTERFACES HOMME-MACHINE

Chapitre 2 : Modèles d'IHM

2.1) Introduction

2.2) Modèle langage

- 1) Principes
- 2) Modèle de Seeheim
- 2.2.3) Évaluation

2.3) Modèle Entrée / sortie

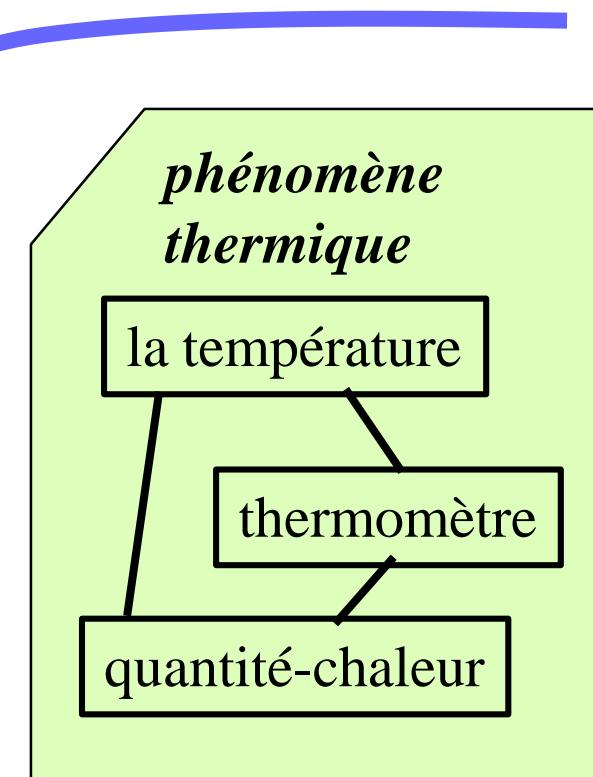
2.4) Modèle Multi-agents

2.2) Modèle langage(1)

Principes

- basé sur l'analogie entre :
 - interaction homme machine
 - dialogue entre 2 individus
- modèle à 3 niveaux :
 - sémantique
 - syntaxique
 - lexical

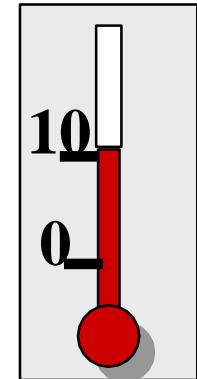
- ✓ L a **sémantique est la signification**
- concepts et savoir faire dans un domaine donné
 - description des classes d'objet



2.2) Modèle langage₍₂₎

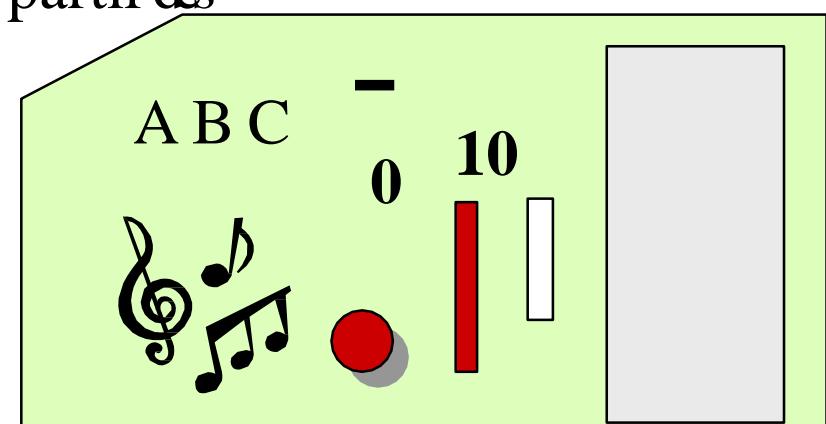
Principes (*suite*)

La **syntaxe** définit la construction de la phrase
structure d'unité syntaxique
élément syntaxique : une unité non décomposable



Le **lexique** définit la production des unités syntaxiques à partir
d'un vocabulaire :

tout ce qu'il est possible de créer à partir des
dispositifs d'entrée et de sortie



A VOUS X

A partir de l'exemple suivant

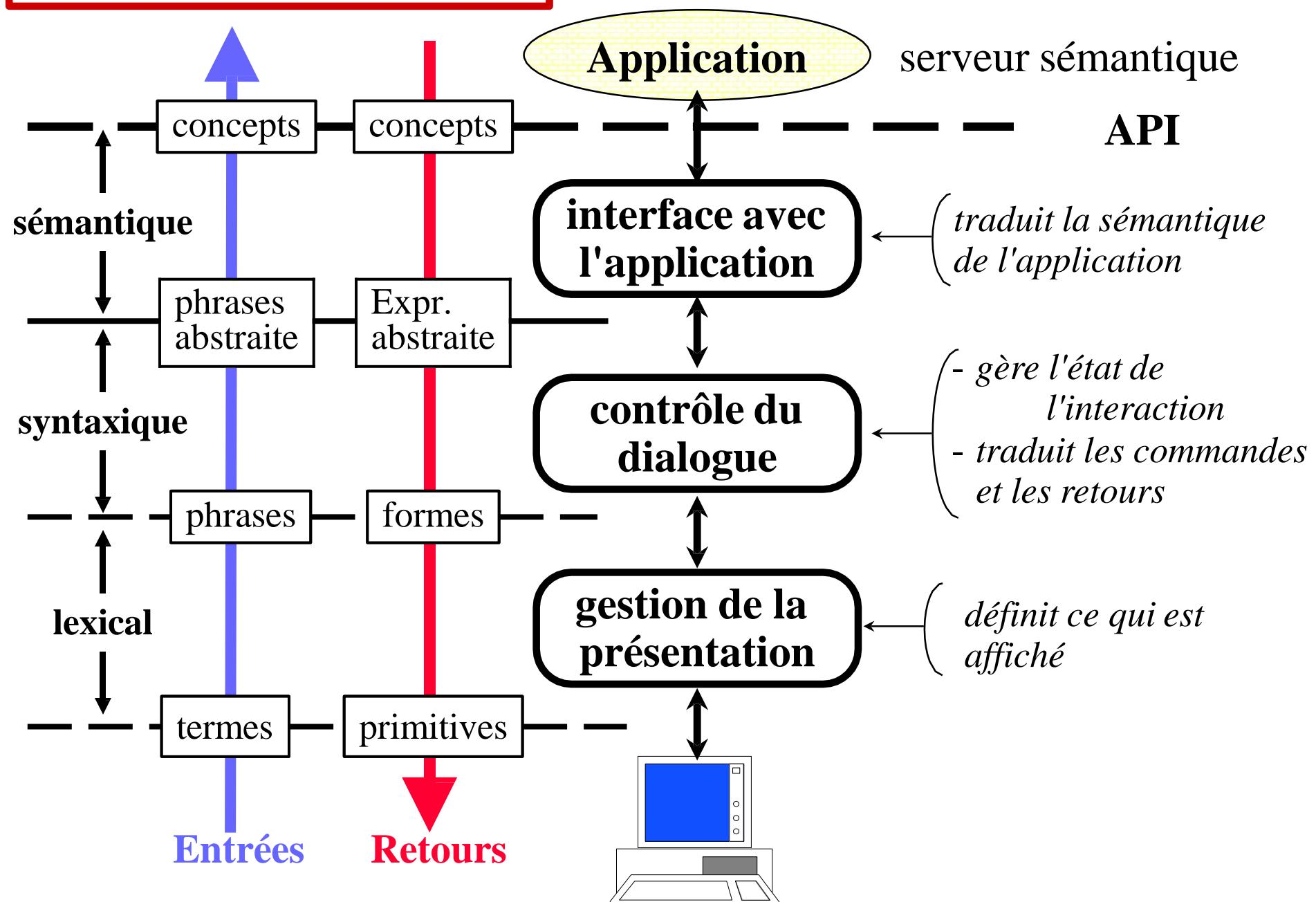
AFFICHER nombre absence injustifiée / «Archibald Haddock» : cours « IHM »

Expliquez :

- ❖ Le lexique
- ❖ La syntaxe
- ❖ La sémantique

2.2) Modèle langage (3)

Modèle de Seeheim



A VOUS X

A partir de l'exemple suivant

Archibald Haddock. : 3 absences injustifiées au cours « IHM »

Expliquez :

- **Le concept**
- **L'expression abstraite**
- **Les formes**
- **Les primitives**



2.2) Modèle langage (4)

Évaluation



apports :

- définition d'un cadre de pensée
- conception itérative des interfaces
- généricité
- généralité

limites :

- priorité à la forme des échanges (*pas à la dynamique*)
- centralisation du traitement des formes (*pb de l'interactivité*)
- découpage du langage en 2 sous langages (*entrée et sortie*)
- imprécision des protocoles de communication

INTERFACES HOMME-MACHINE

Chapitre 2 : Modèles d'IHM

- 1) *Introduction*
- 2) *Modèle langage*

2.3) Modèle Entrée / Sortie

- 2.3.1) Introduction
- 2.3.2) Vue en couches
- 2.3.3) évaluation

2.4) Modèle Multi-agents

2.3) Modèle Entrée / Sortie (1)

2.3.1) Introduction

Accent sur l'**échange** et la **transformation** d'information.

la forme des échanges et la nature des informations

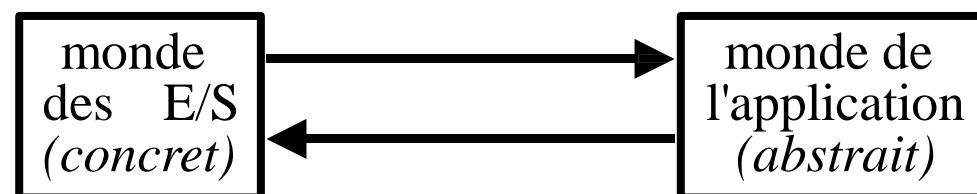
- le découpage fonctionnel.

- un empilement de machines abstraites

- une organisation de processus communiquant.

Pour **cacher** la complexité du monde physique à l'application

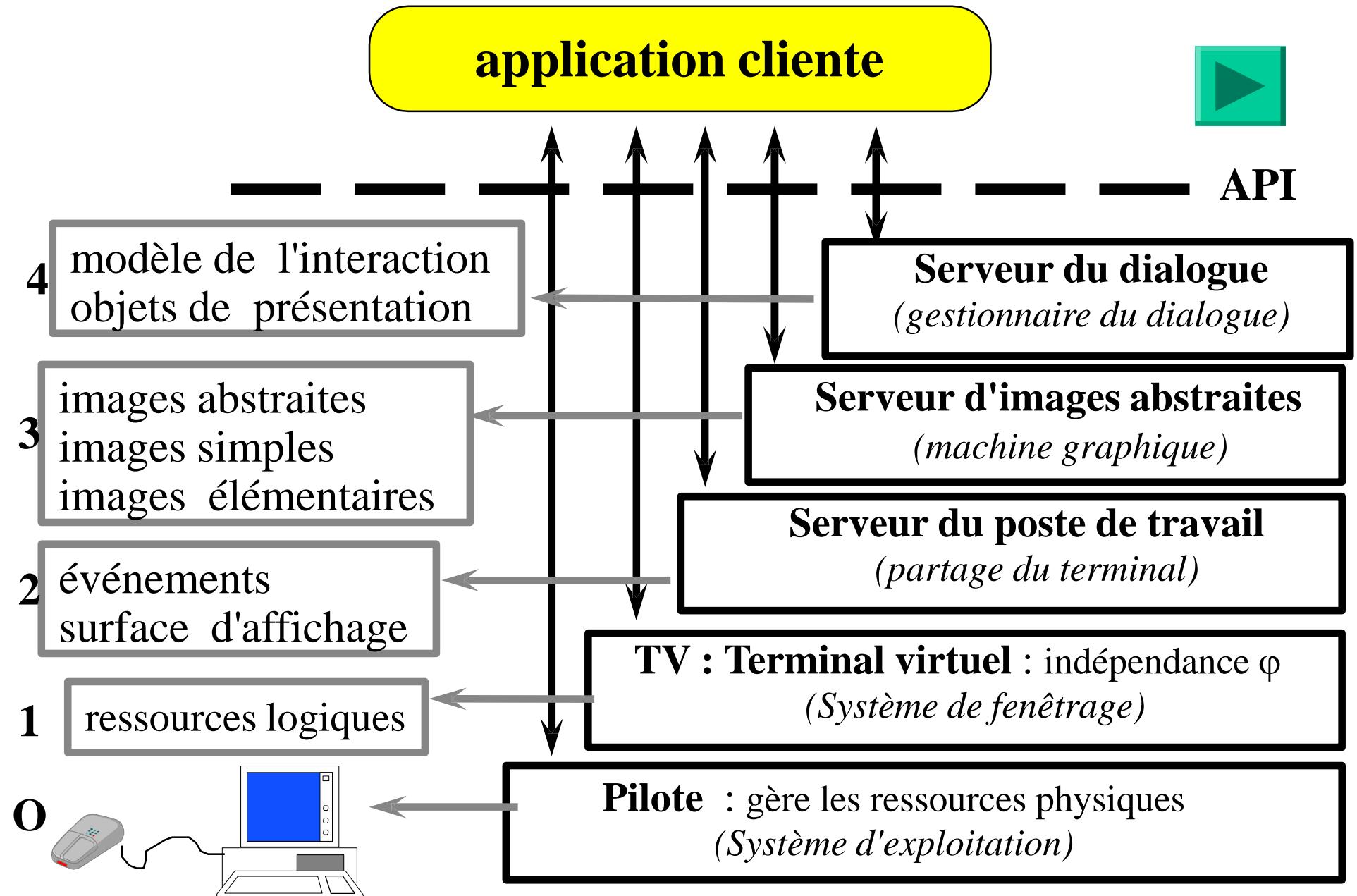
on **stratifie** l'interface en niveaux, faisant passer



ce modèle s'inspire du modèle ISO

2.3) Modèle Entrée / Sortie (2)

2.3.2) Vue en couches



2.3) Modèle Entrée / Sortie (3)

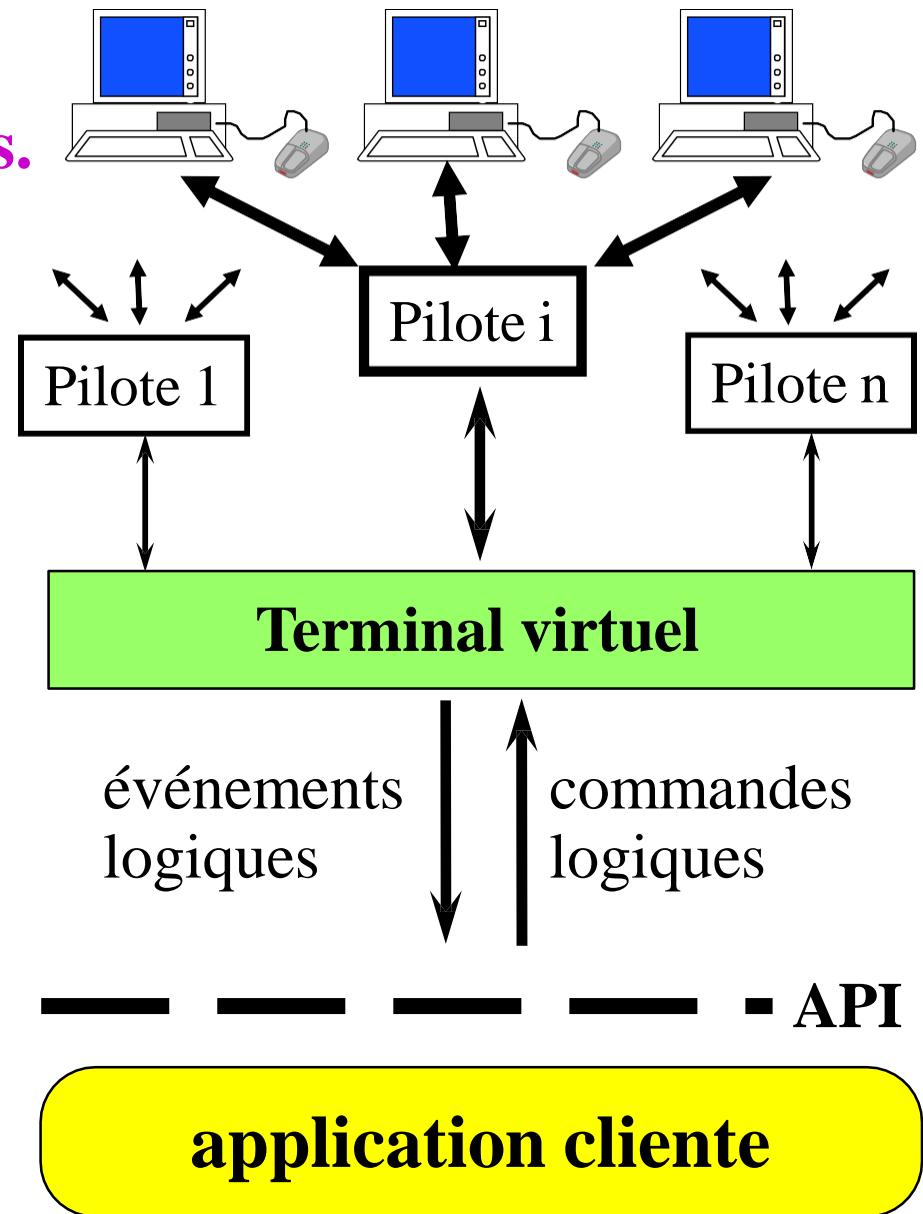
Niveau 0 : dispositifs physiques.

Les pilotes gèrent des **classes** de terminaux physiques.

Niveau 1 : dispositif logique

assure **l'indépendance** de l'application par rapport aux dispositifs physiques

2.3.2) Vue en couches



2.3) Modèle Entrée / Sortie (4)

2.3.2) Vue en couches



Niveau 2 : Serveur du poste de travail

gère les **ressources**

gère le **partage** des ressources

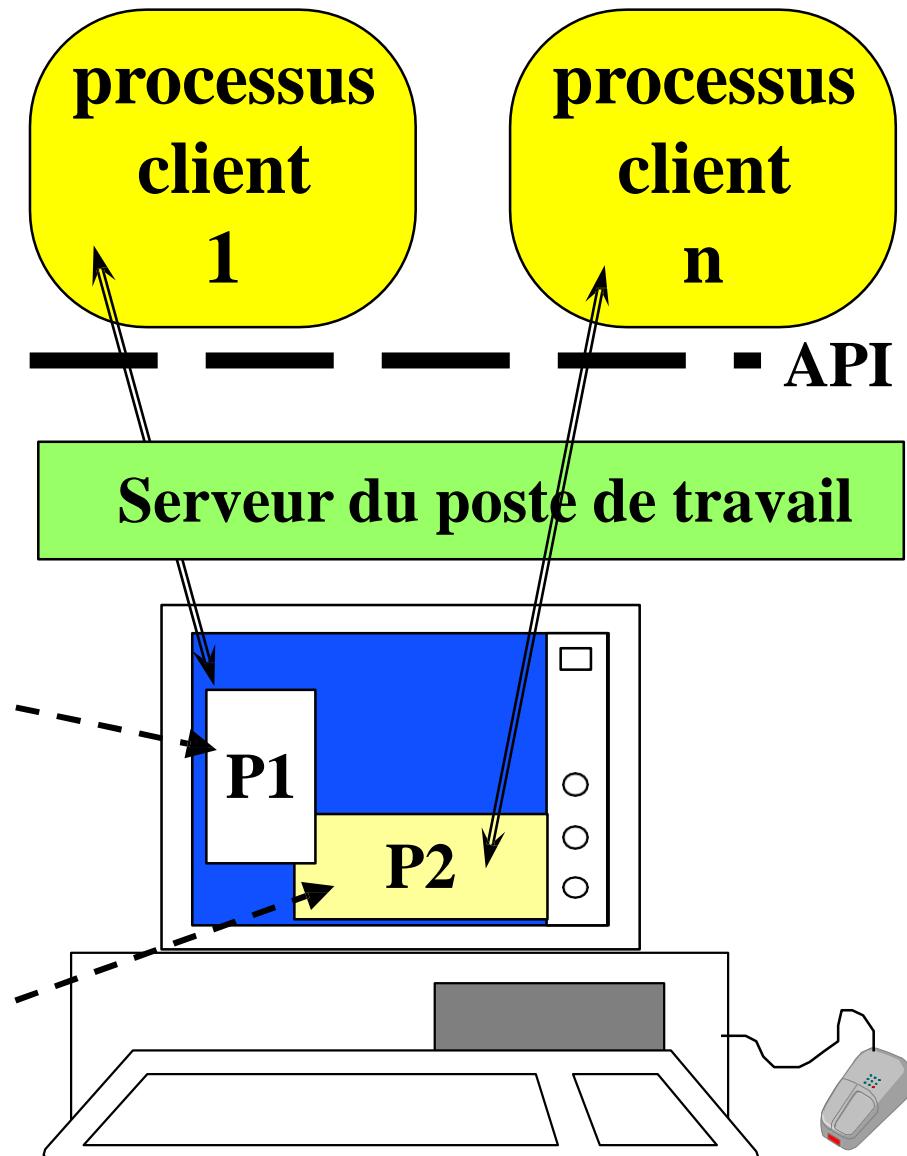
gère les Terminaux virtuels (TV)
processus client 1 **fenêtre**

1 **utilisateur** dialogue **un**
seul **processus** à la fois
fenêtre active

TV du processus 1^{er} plan :

fenêtre active
+ clavier
+ souris
+ HP,

TV d'un processus 1^{er} plan :
fenêtre inactive



2.3) Modèle Entrée / Sortie (5)

Niveau 3 : serveur d'images abstraites

3 niveaux de représentation :

l'application possède son modèle de **représentation interne** des objets.

un modèle **intermédiaire** qui prend en charge les actions utilisateurs au niveau syntaxique

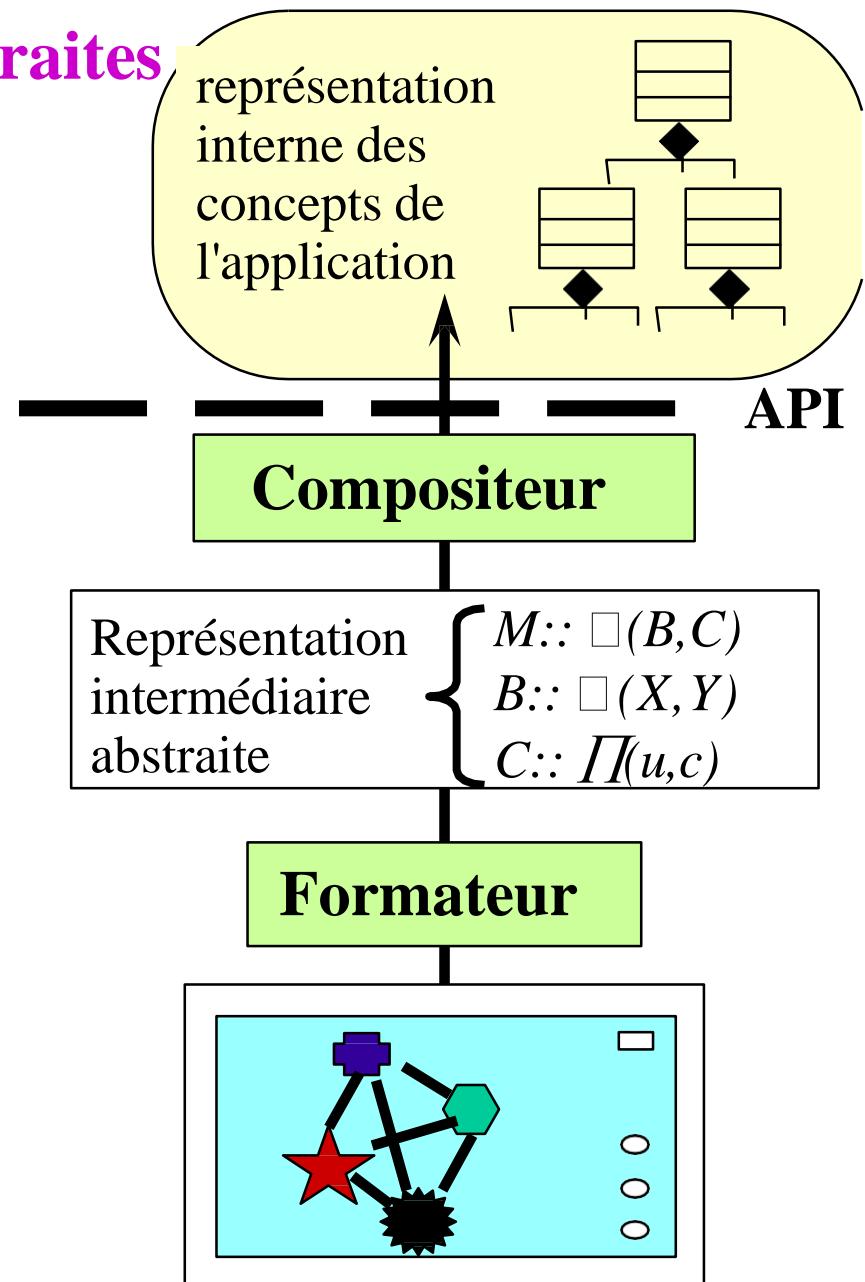
modèle de représentation **interne au poste de travail**.

2 interfaces :

Compositeur : génère & structure les objets abstraits.

Formateur : génère les représentations réelles

2.3.2) Vue en couches



2.3) Modèle Entrée / Sortie (6)

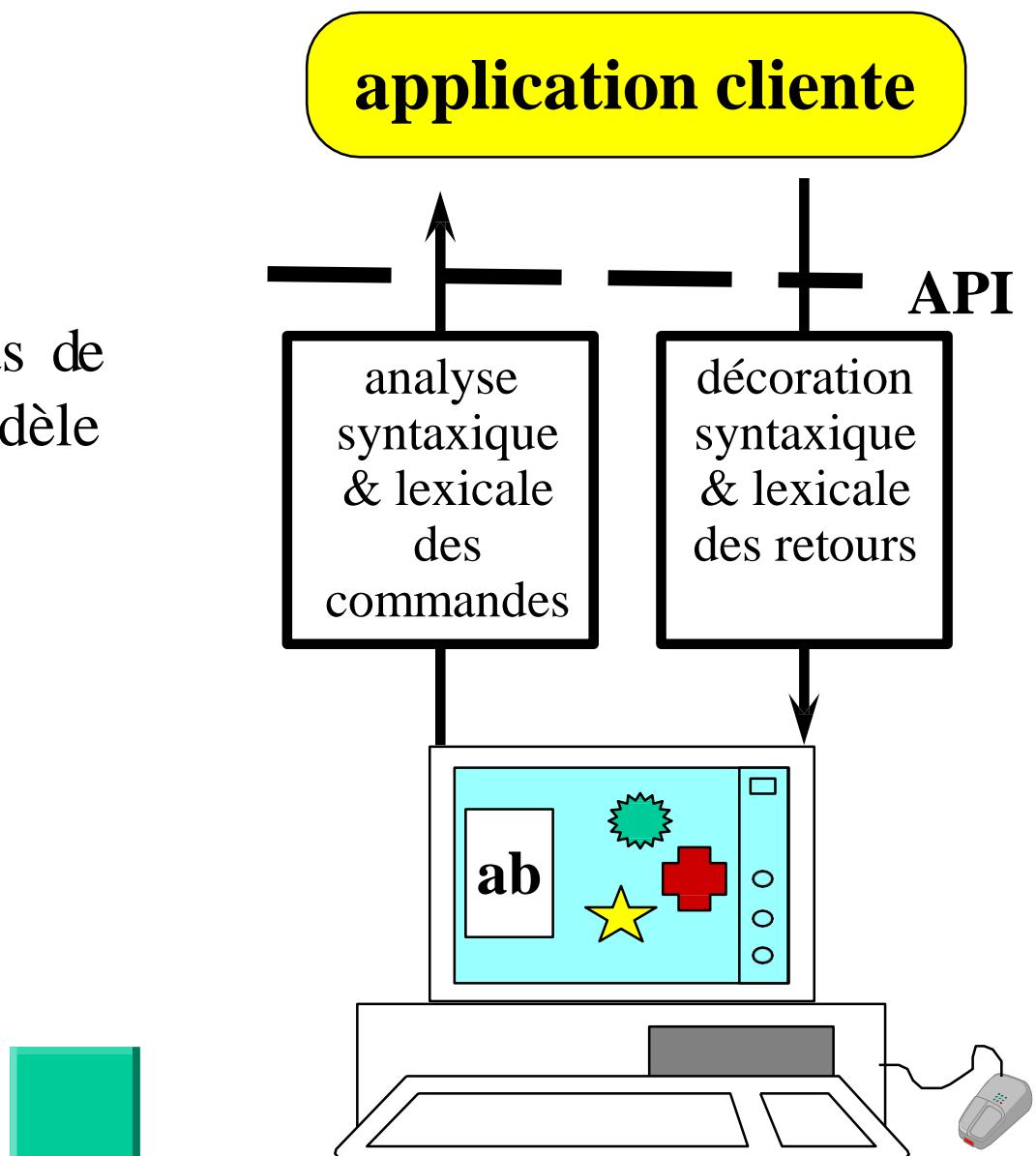
Niveau 4 : serveur du dialogue

décharge l'application des problèmes de l'interface

contrôle un ensemble d'objets de présentation au moyen du modèle de l'interaction.

Les échanges se font au niveau **sémantique** :
commandes
retours

2.3.2) Vue en couches



2.3) Modèle Entrée / Sortie (7)

2.3.3) évaluation



Apports

il fournit **une vision globale** des transformations appliquées aux informations échangées;

la **structure en couche** facilite la compréhension du rôle des différents niveaux de traitement;

la **structure modulaire** se rapproche de la dynamique des échanges et des problèmes de réalisation.

limites

Les **niveaux d'abstraction** sont de trop bas niveau pour construire efficacement un dialogue complexe avec la notion d'événements, d'actions parallèles, de synchronisations, etc.

INTERFACES HOMME-MACHINE

Chapitre 2 : Modèles d'IHM

- 1) *Introduction*
- 2) *Modèle langage*
- 3) *Modèle Entrée / Sortie*

➤ 2.4) Modèle Multi-agents (*Objet*)

- 2.4.1 principes**
- 2.4.2 agents objets**
- 2.4.3 modèle PAC**
- 2.4.4 modèle MVC**
- 2.4.5 synthèse**

2.4) Modèle multi-agents (1)

2.4.1) principes

Types de dialogue :

synchrone (*question réponse ou menu à choix unique*)
le contrôle est simple.

asynchrone (*interface événementiel*)
possibilités de choix des utilisateurs sont très nombreuses
plusieurs actions peuvent être lancées en même temps
les réponses arrivent en désordre, etc.
le contrôle est beaucoup plus complexe

2.4) Modèle multi-agents (2)

2.4.1) principes

Le modèle multi-agents

structure un système interactif en un **ensemble d'agents spécialisés** qui

réagissent à des événements
produisent des événements

Un événement est un stimulus qui :

appartient à une **classe d'événements**

véhicule de **l'information**

est produit par un **émetteur**

est détecté par des **récepteurs**

2.4) Modèle multi-agents (3)

2.4.1) principles

Un agent :

objet complexe

traitement de l'information.

des émetteurs

des récepteurs

une **mémoire** à 2 niveaux

pour enregistrer les événements détectés

pour mémoriser un état

un **filtre** d'événements

un processeur cyclique

traite une ou plusieurs classes d'événements.

*les événements sont traités **séquentiellement***

2.4) Modèle multi-agents (4)

2.4.1) principes

Lorsqu'un **émetteur produit un événement**,

les **récepteurs sensibles**

sont activés

l'enregistrent

le traitent

le **traitement** se traduit généralement par :

un changement d'état de l'agent

l'émission de nouveaux événements

Le modèle multi-agents

une architecture fortement **modulaire**,

des traitements exécutés en **parallèle**

une communication par **événements**

Le modèle multi-agents est à rapprocher du modèle à dit

2.4) Modèle multi-agents (5)

2.4.2) agents ↔ objets

catégorie d'agent

classe d'objets

mémoires

attributs

répertoire d'instructions du processeur de l'agent

methods

sémantique des instructions est spécifiée par des contraintes
une de ces contraintes est le filtre détecteur d'événement.

émission événement :

l'envoi d'un message par un autre agent
une intervention de l'utilisateur

2.4) Modèle multi-agents (6)

2.4.2) agents ↔ objets

Agent interactif

(ou *objet interactif* ou *intéacteur*)

médiateurs entre le **monde abstrait** du système
 et le **monde réel** de l'utilisateur

double comportement **interne** : vis à vis de l'application
 externe : vis à vis de l'utilisateur

évoluent en fonction des messages qu'ils reçoivent

communiquent avec les autres objets par messages

un événement «**actions de l'utilisateur sur un objet**»

message(s) : objet-IHM objet-NA

sélection de **méthode(s)**

comportement

Modèles multi-agent : PAC, MVC,

2.4) Modèle multi-agents (7)

2.4.3) modèle PAC

Auteur : Joëlle COUTAZ

Objectif : intégrer

la **structuration de la communication** du modèle langage :

- niveau lexical,
- niveau syntaxique,
- niveau sémantique

la **puissance et la souplesse** du modèle multi-agent

- modularité
- parallelisme
- meilleure prise en compte de l'utilisateur
- facilité du travail du concepteur

l'**architecture en couche** du modèle Entrée /Sortie

2.4) Modèle multi-agents (8)

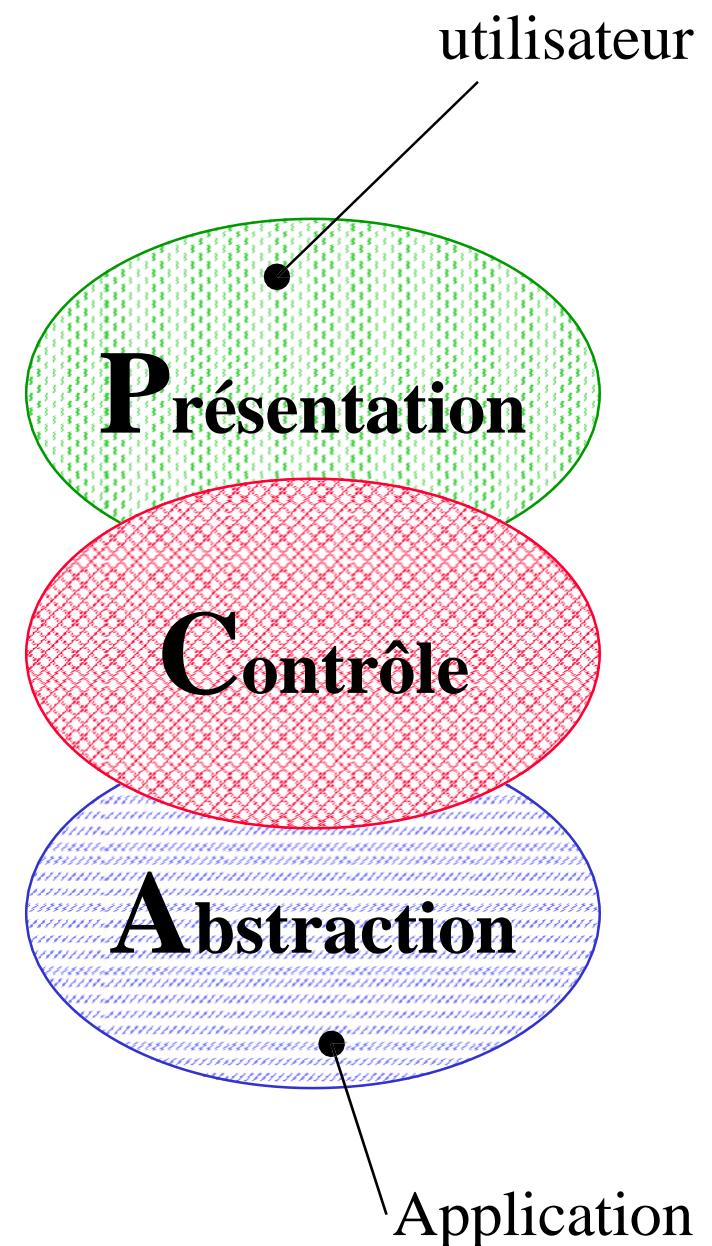
agent PAC

l a **présentation** qui définit l'image du système et son comportement vis à vis de l'**utilisateur**

l'**abstraction** qui définit les concepts de l'application et les fonctions du système, vu des autres **constituants logiciels**

l e **contrôle** maintient la cohérence entre l'abstraction et la présentation

2.4.3) modèle PAC



2.4) Modèle multi-agents (8)

Objet Interactif Élémentaire (OIE)

1 a présentation

en entrée : les actions de l'utilisateur

en sortie : les primitives graphiques

l'abstractio

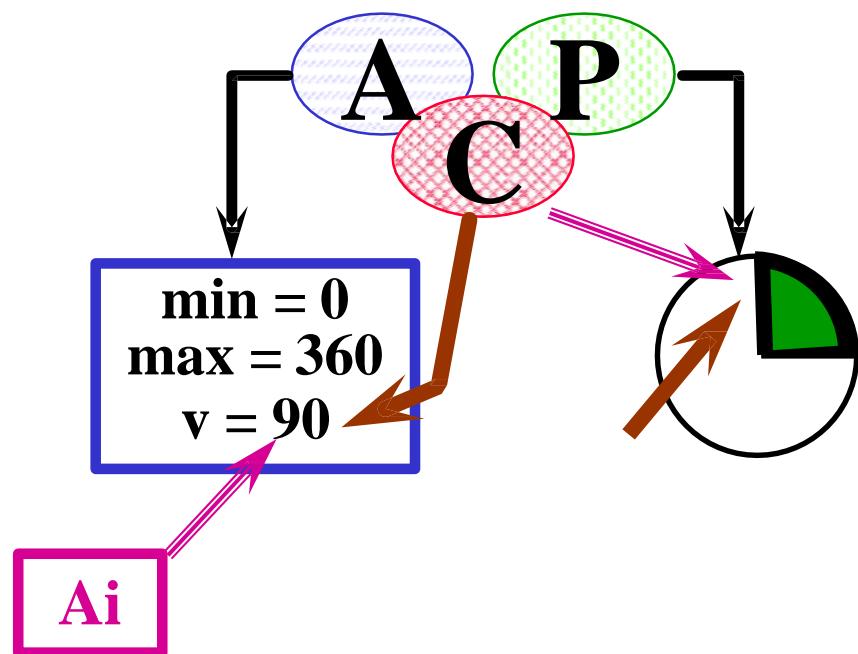
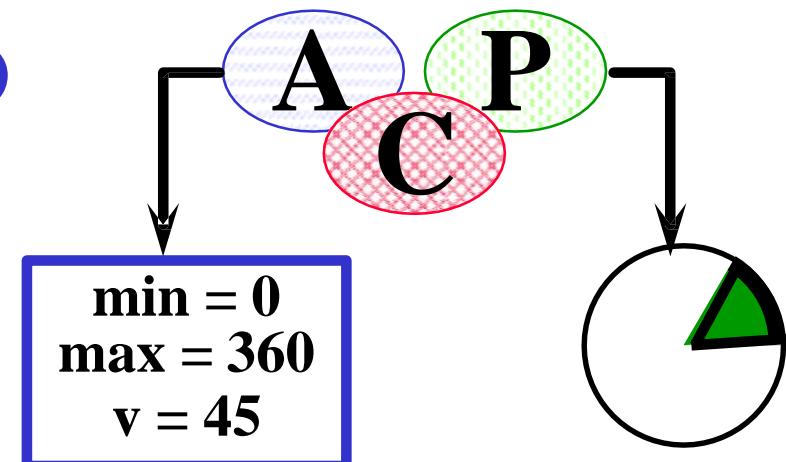
- n les valeurs v, min et max.
le domaine (réels)

1 e contrôle maintient la cohérence si :

l'utilisateur modifie au niveau de P

un événement provenant d'un autre agent
modifie la valeur dans A

2.4.3) modèle PAC



A VOUS X

Dans l'exemple précédent

Quel est le concept ?

Quels sont les évènements

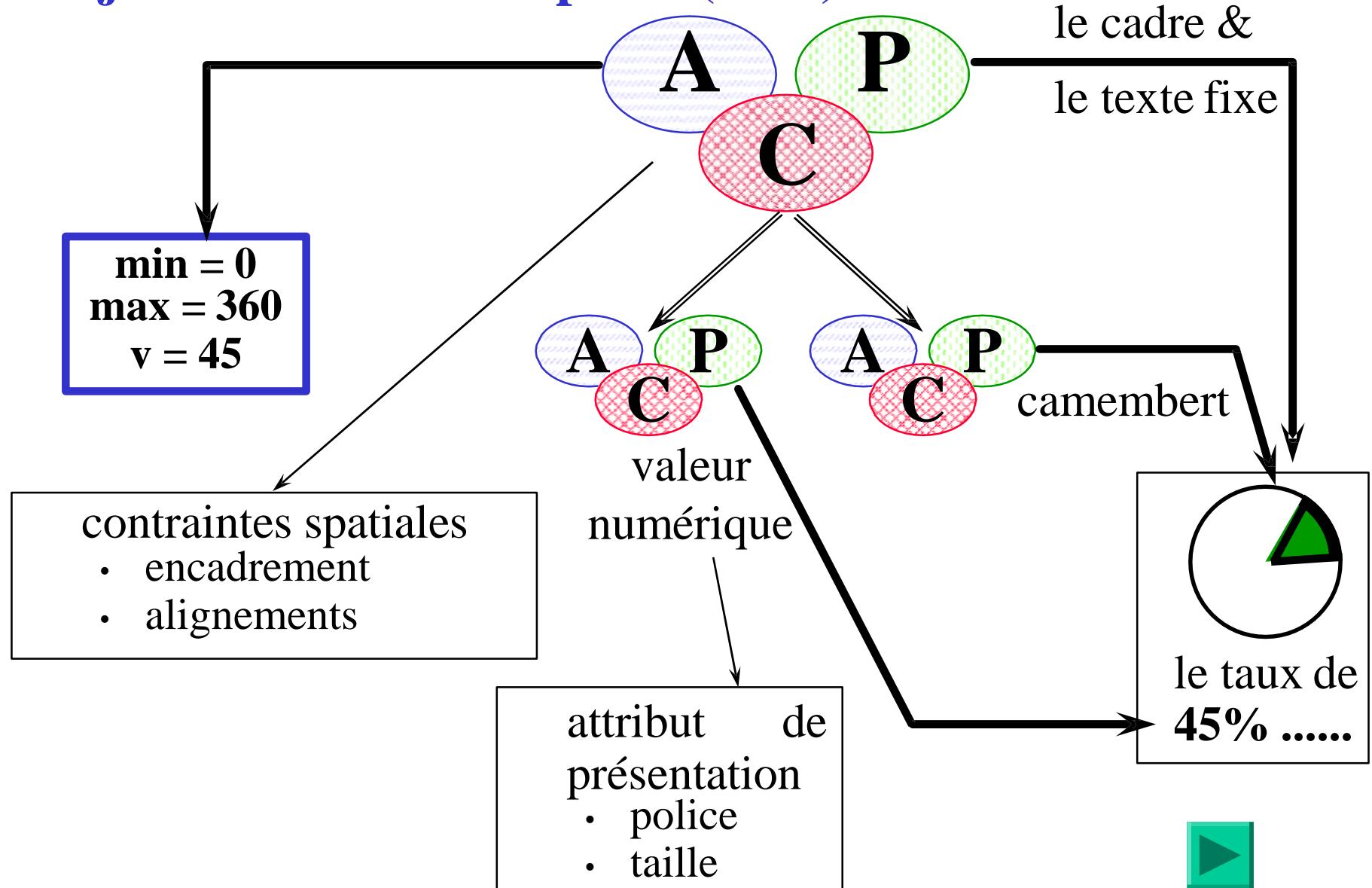
Quels sont les états

Quels sont les traitements

2.4) Modèle multi-agents (9)

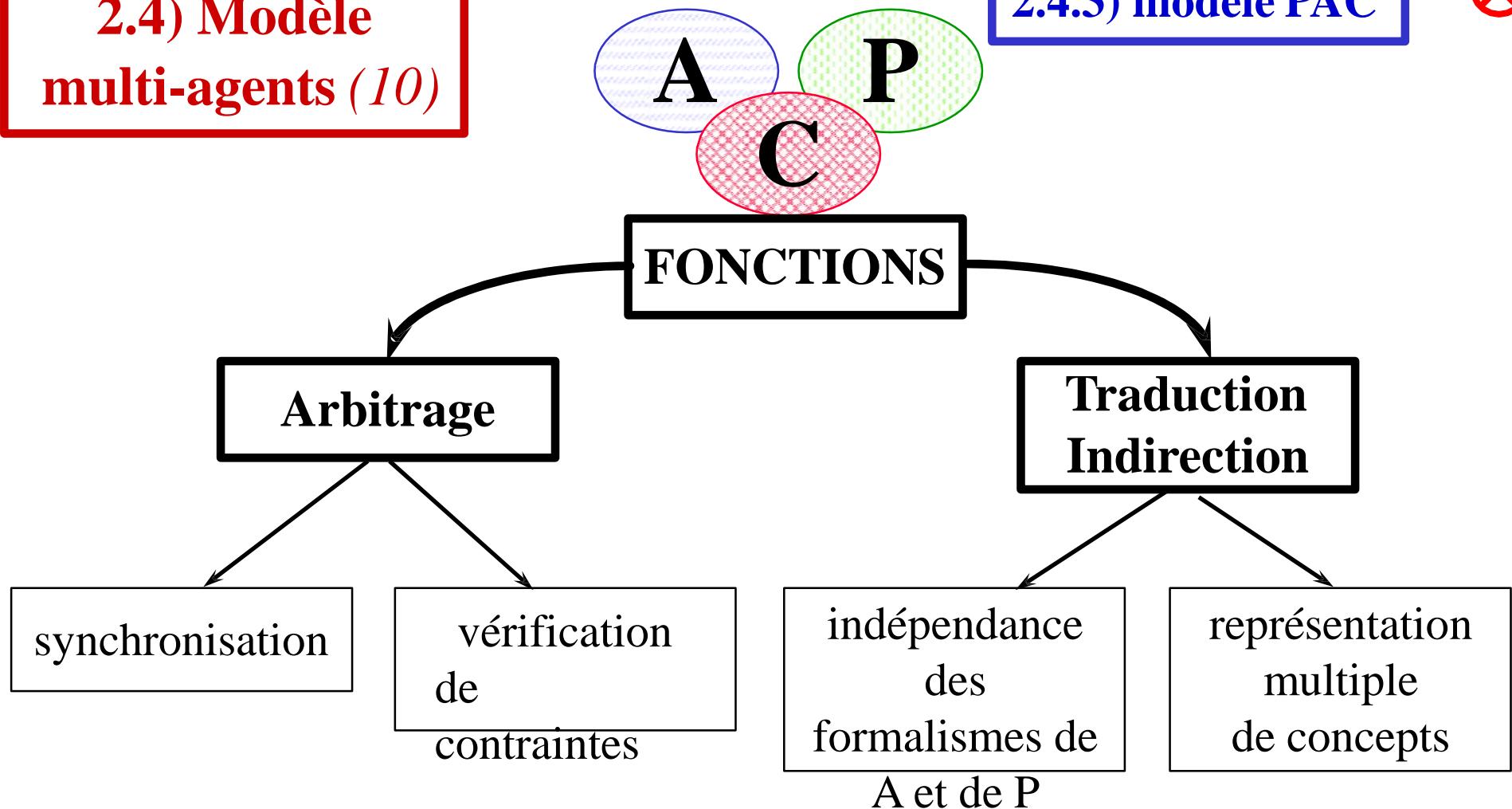
2.4.3) modèle PAC

objets interactifs composés (OIC)



2.4) Modèle multi-agents (10)

2.4.3) modèle PAC



Contrôle

un **pont** entre P et A

peut associer **plusieurs présentations** à une abstraction

décide de l'**ordre** d'activation des agents, de traitement des événements

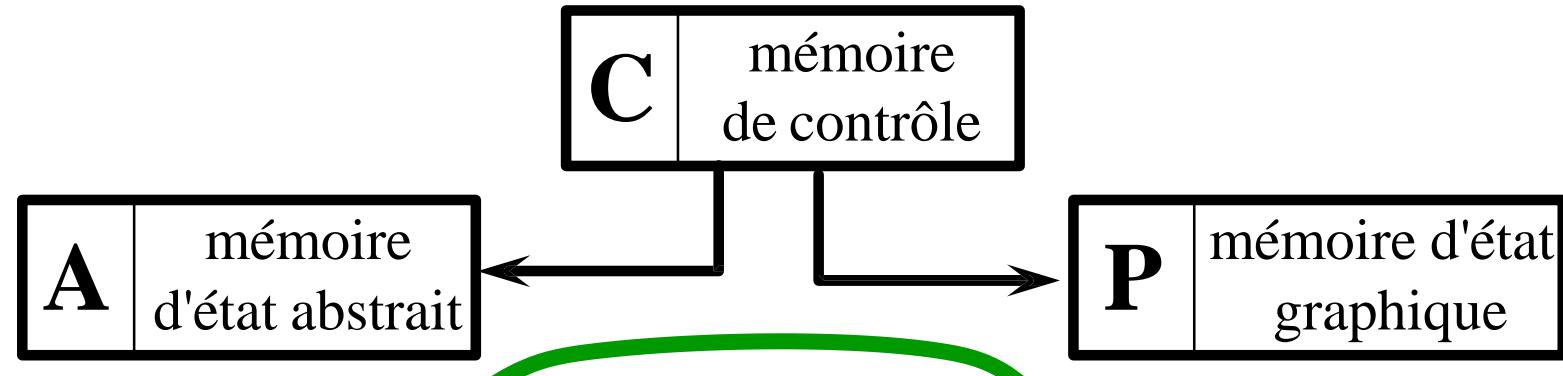
gère l'information **contextuelle**

2.4) Modèle multi-agents (11)

OIE

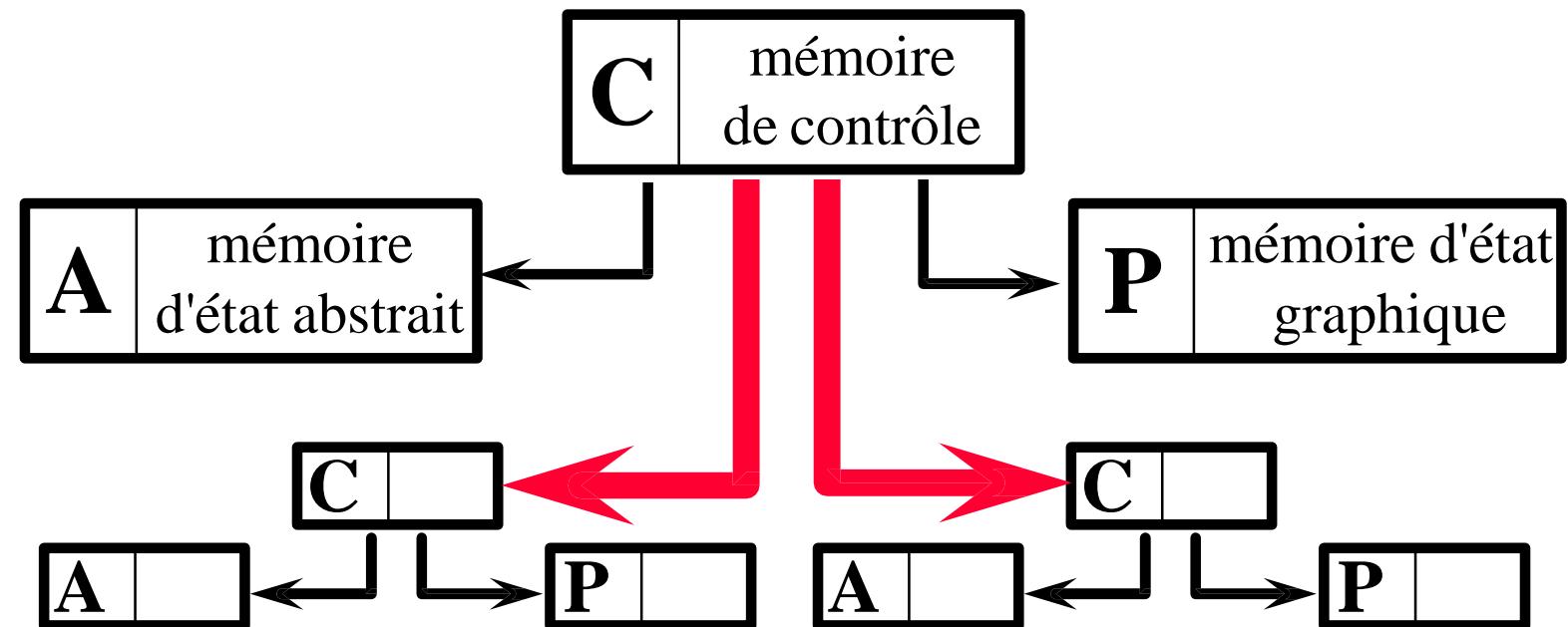
événements

2.4.3) modèle PAC



OIC

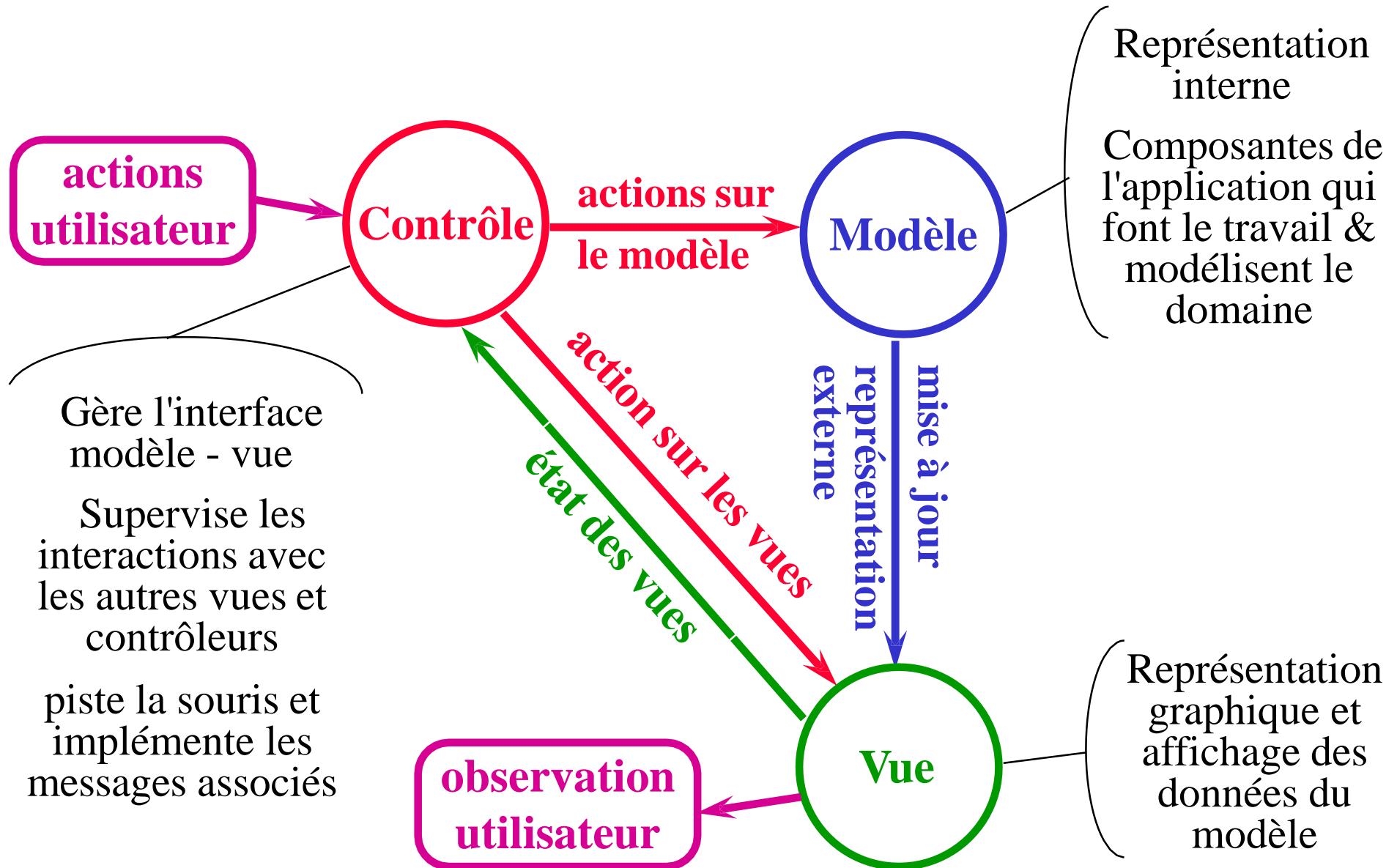
événements



2.4) Modèle multi-agents (12)

2.4.4) modèle MVC

Model - View - Contrôleur (SMALLTALK, JAVA, Visual C++, ...)



A VOUS X

**Modèle MVC : En reprenant l'exemple de PAC,
indiquez :**

Les contrôles

Les vues

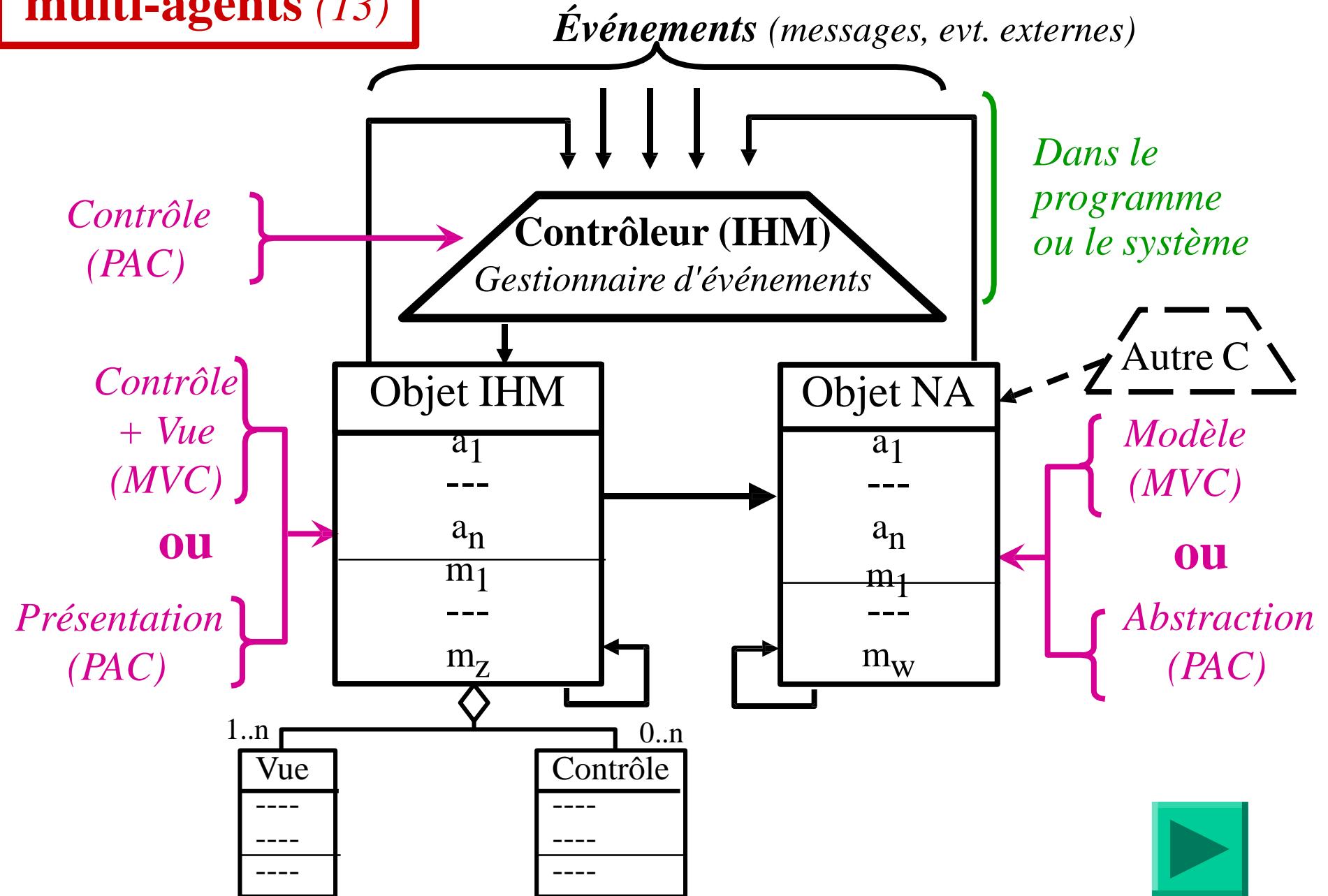
Expliquez le modèle

A VOUS X

Quelle est la principale différence entre PAC et MVC?

2.4) Modèle multi-agents (13)

2.4.5) Implémentation objet possible



2.4) Modèle multi-agents (14)

2.4.5) synthèse

modularité :

indépendance de P et A (*toute communication passe par C*)

les modifications applicables à tous les niveaux (OIE & OIC).

la répartition : sémantique abstraction
lexique & syntaxe présentation

dialogue à plusieurs fils d'activité

interactions dirigées par l'utilisateur

E/S peuvent se réaliser simultanément et en //

action simultanée sur la présentation de # concepts

abstraction et récursivité

modèle récursif applicable à tous les niveaux d'abstraction.

les échanges se font au niveau d'abstraction voulu