Modèles pour l'Interaction Homme-Machine

Tronc commun RICM3
2006-2007

Renaud Blanch
IIHM - CLIPS-IMAG - UJF
mailto:renaud.blanch@imag.fr
http://iihm.imag.fr/blanch

3. Architecture logicielle

3.0 Modèles de référence :

- **modèle de Seeheim**
- **modèle de l'Arch**

3.1 Modèles de référence à agents :

- **modèle MVC**
- **modèle ALV**
- **modèle PAC**

3.2 Modèle de référence hybride :

- **modèle PAC-Amodeus**

Introduction

Constat :
• conception d'IHM difficile, par conséquent itérative (itérativité implique modifiabilité du logiciel)
• complexité et taille croissante des IHM
• outils de développement des IHM imparfaits (boîtes à outils, squelettes d'application, générateurs d'interface)

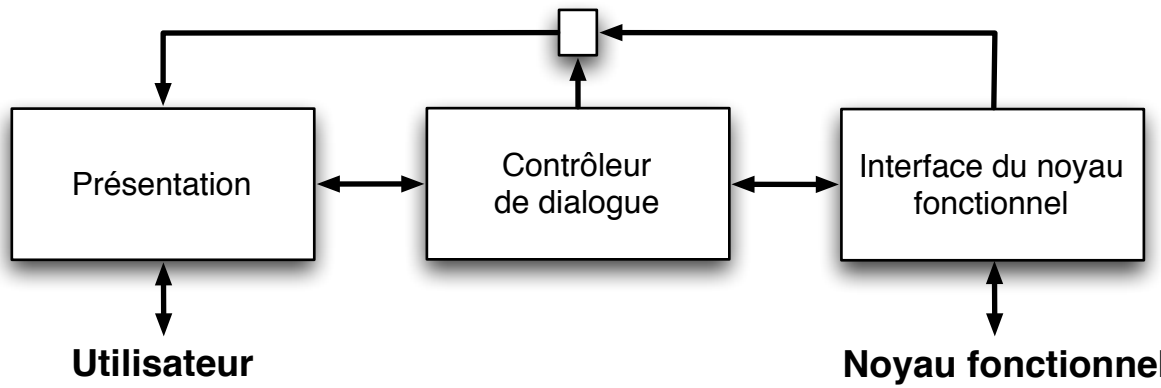
Conséquence :
On a besoin d'un cadre de pensée.

L'architecture est abstraite, elle décrit des composants et leurs relations sans présumer de leur réalisation.
Il existe des architectures éprouvées qui sont des modèles de référence.

3.0 Modèles de référence

Modèle fondamental

Modèle de Seeheim [Pfaff, 1985]



Modèle de Seeheim [Pfaff, 1985]

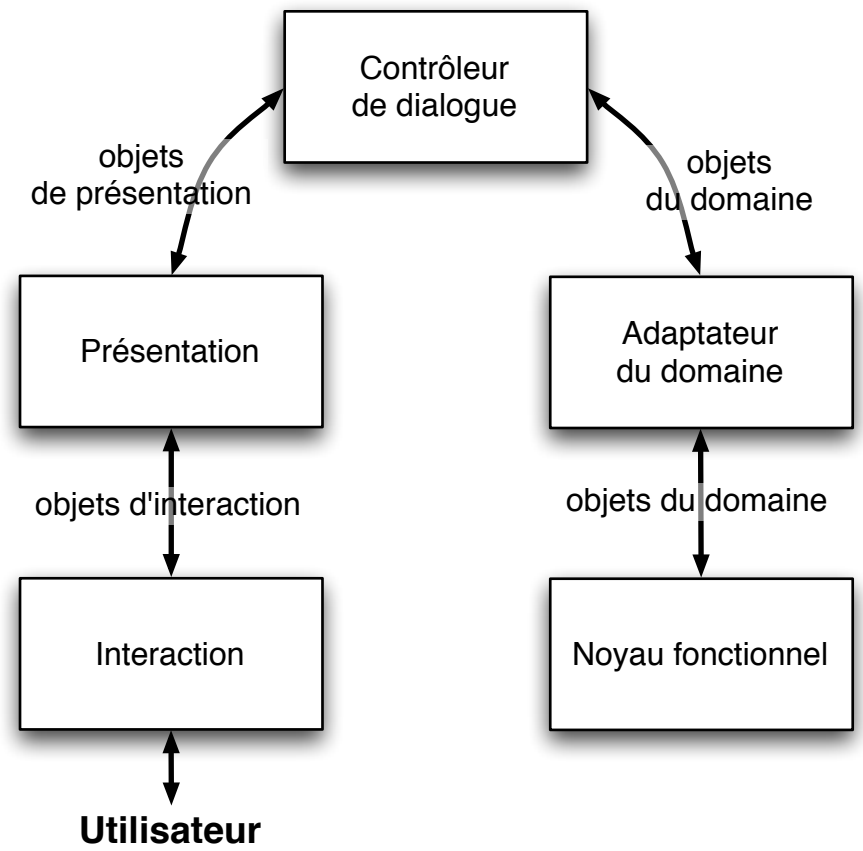
Le **noyau fonctionnel**
manipule les objets du domaine d'application.

L'**interface du noyau fonctionnel**
décrit la sémantique de l'application du point de vue de l'interface utilisateur.

La **présentation**
définit le comportement du système tel qu'il est perçu par l'utilisateur et tel qu'il manipule.

Le **controleur de dialogue**
agit comme médiateur entre l'interface du noyau fonctionnel et la présentation.

Modèle de l'Arch [1992]



Modèle de l'Arch [1992]

Arch est une version raffinée de Seeheim
qui tient compte de l'apparition des boîtes à outils.
Les pieds de l'arche représentent les éléments préexistants : noyau fonctionnel d'une part et boîte à outils d'interface d'autre part.

La **présentation** et l'**adaptateur du noyau fonctionnel** permettent l'adaptation entre les 3 autres composants.

Ces adaptations ne sont pa toujours nécessaires :
le méta-modèle **Slinky** autorise alors leur disparition ou leur fusion dans les autres éléments de l'arche.

3.1 Modèles de référence à agents

Principe :
Un système interactif est constitué d'une collection d'unités de calcul spécialisées (agents).

Un agent :
- a un état ;
- a une expertise ; et
- est capable d'émettre et de réagir à des événements.

Un interacteur est un agent en contact direct avec l'utilisateur.

Modèle MVC [smalltalk, 1981]

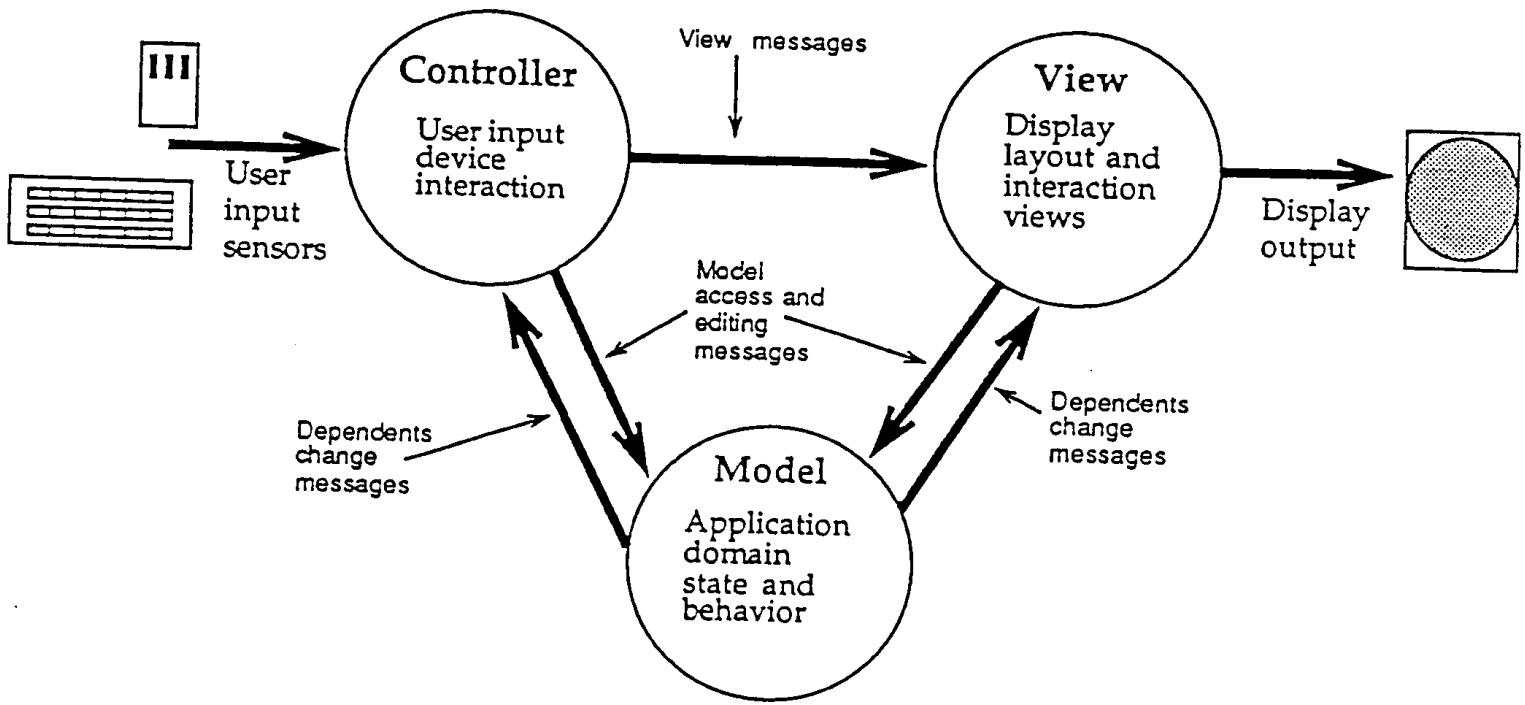


Figure 1: Model-View-Controller State and Message Sending

Modèle MVC [smalltalk, 1981]

Un agent MVC est composé de trois facettes réalisées par des objets :
le **modèle** (*Model)*, la **vue** *(View)* et le **contrôleur** *(Controller).*

Les communications entre la vue et le contrôleur ne peuvent normalement se faire sans passer par le modèle qui est garant de la cohérence de l'état de l'agent.

Le **modèle**
maintient son état et **notifie** de ses modifications.

Le **contrôleur**
écoute l'utilisateur et **demande** des modifications au modèle.

La **vue**
écoute les notifications et **interroge** le modèle pour le représenter correctement.

Modèle MVC [smalltalk, 1981]

Limitations :
• qui gère la sélection ?
• besoin d'une communication vue/contrôleur pour gérer le *feed-back* ...

Réponse :
La vue et le contrôleur ont souvent été réunis au sein d'une même facette dans les modèles postérieurs (Java/SWING procède de la sorte).

Attention :
MVC a été réutilisé dans le domaine des applications Web, mais il ne s'agit pas exactement du même modèle.

Modèle ALV [Hill, 1992]

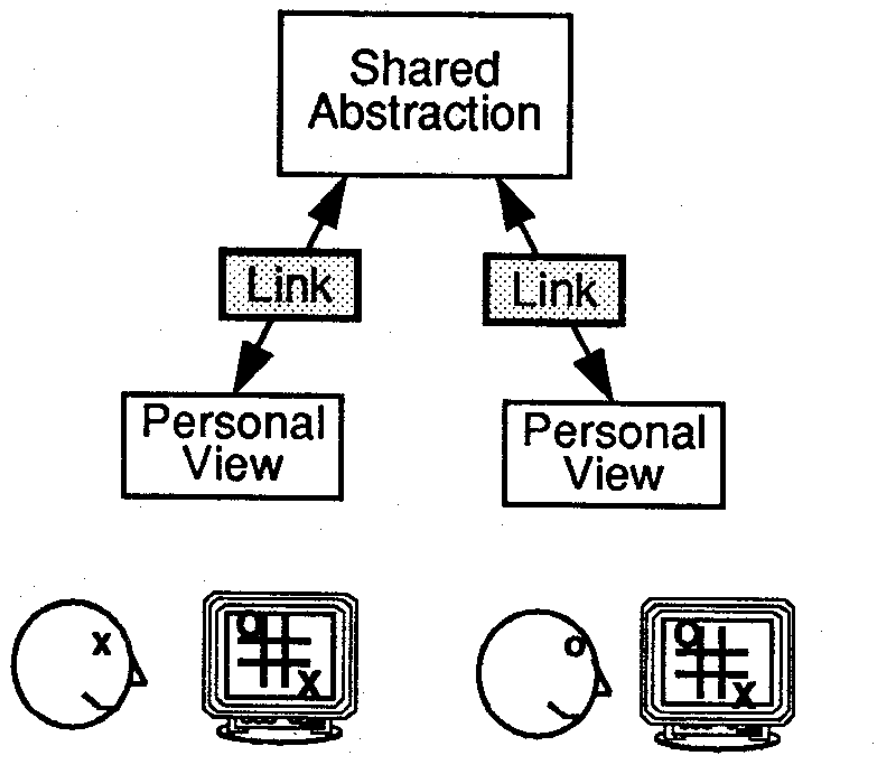


Figure 4. Abstract structure of two-user game.

Modèle ALV [Hill, 1992]

Un agent ALV est composé de trois facettes :
• l'**abstraction** *(Abstraction)* (le modèle de MVC) ;
• la **vue** *(View)* (la vue et le contrôleur de MVC) *;* et
• les **liens** *(Links)* qui sont l'expression des dépendance entre abstraction et vue*.*

Les liens sont décrit dans un langage dédié à contrainte.
Les abstractions et les vues sont réalisées par des objets.

Modèle PAC [Coutaz, 1987]

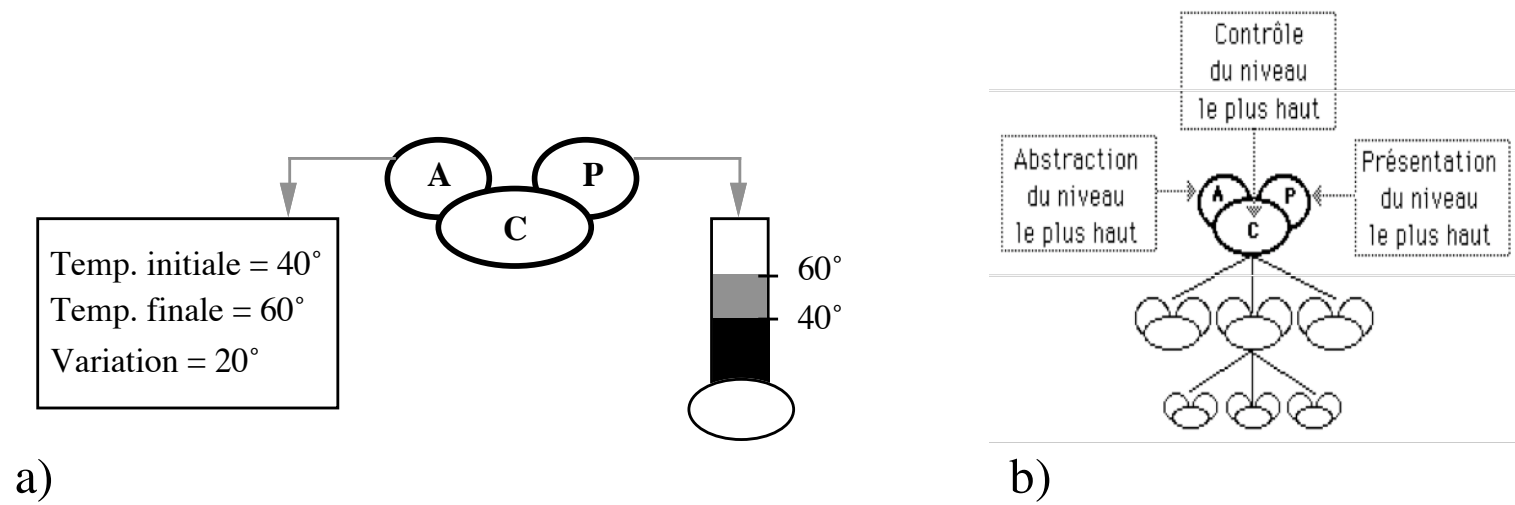


Figure 7.4. *a) Un exemple d'agent PAC. b) Le modèle d'architecture PAC. Les traits pleins représentent des échanges entre agents de niveaux d'abstraction distincts.*

Modèle PAC [Coutaz, 1987]

Un agent PAC est composé de trois facettes réalisées par des objets :

• l'**abstraction** (le modèle de MVC) ;

• la **présentation** (la vue et le contrôleur de MVC) *;* et

• le **contrôle** qui exprime les dépendance entre abstraction et présentation et qui gère les échanges avec les autres agents.

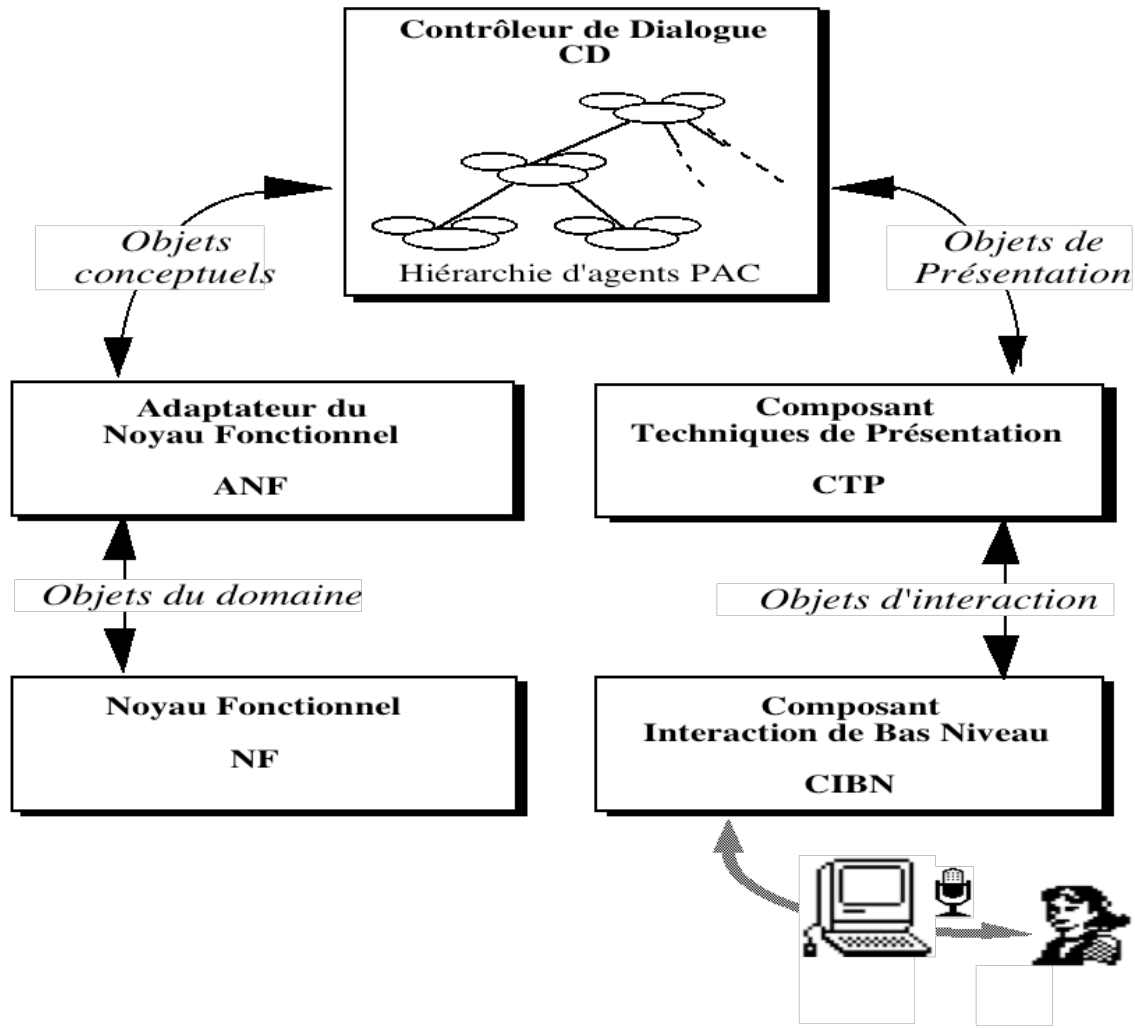
Modèle PAC [Coutaz, 1987]

Le **contrôle**
notifie le **modèle** lorsque les manipulations de la **présentation** par l'utilisateur le nécessitent ; et
notifie la **présentation** lorsque les modifications du **modèle** le réclame.

Le **contrôle**
notifie les facettes **contrôle** des autres agent PAC de la hiérarchie si besoin.

3.2 Modèles de référence hybride

Modèle PAC-Amodeus [nigay, 1991]



Modèle PAC-Amodeus [nigay, 1991]

Le modèle **PAC-Amodeus**
utilise **PAC** pour raffiner le **contrôleur de dialogue** du modèle de l'arche.

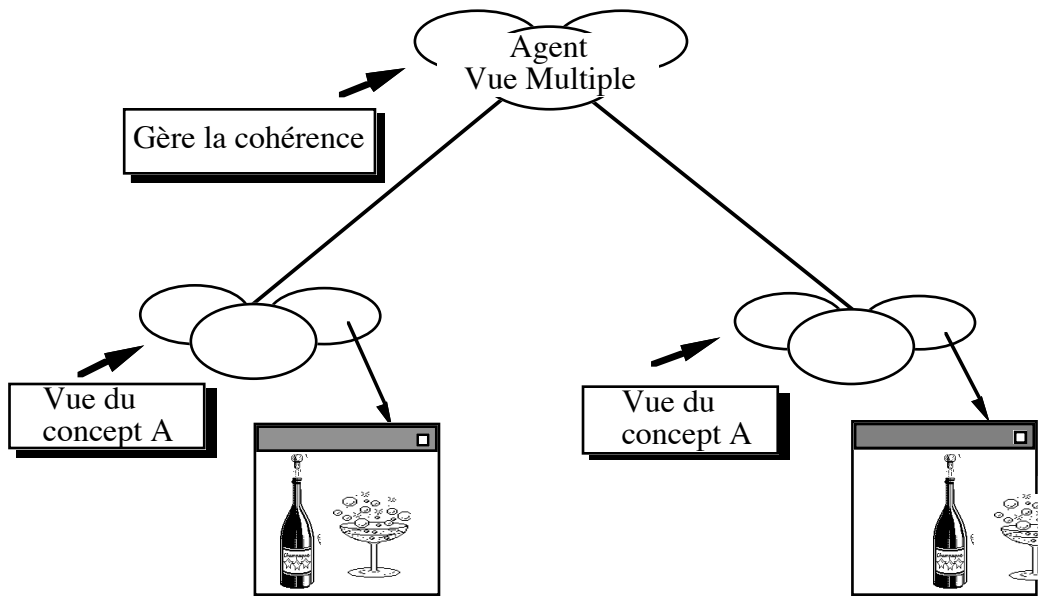
Le modèle **PAC-Amodeus**
propose un ensemble de **règles heuristiques** pour guider la structuration en hiérarchie d'agents PAC.

Modèle PAC-Amodeus [nigay, 1991]

règle 1 :
Une fenêtre qui sert de support à un espace de travail est modélisée par un agent.

Modèle PAC-Amodeus [nigay, 1991]

règle 2 :
Les vues multiples d'un même concept sont gérées par un agent "vue multiple" chargé de maintenir la cohérence entre les vues.



Modèle PAC-Amodeus [nigay, 1991]

règle 3 :
Une palette est modélisée par un agent.

règle 4 :
Une barre de menu est modélisée par un agent.

règle 5 :
Une zone d'édition est modélisée par un agent.

règle 6 :
Un concept complexe est modélisée par un agent.

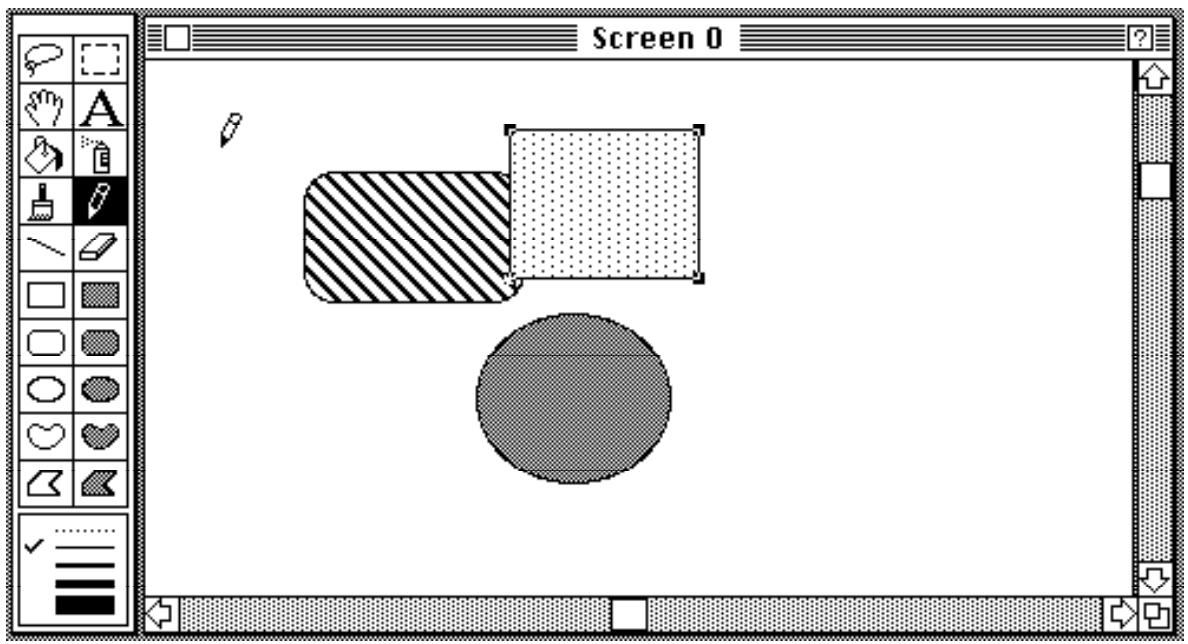
Modèle PAC-Amodeus [nigay, 1991]

règle 7 :
Si une fenêtre “espace de travail” permet d’ouvrir une autre fenêtre “espace de travail” sur une autre instance du même concept, ces deux agents sont modélisés comme fils d’un même père.

règle 8 :
Si la nouvelle fenêtre représente plus de détails sur l’un des concepts, l’agent qui modélise cette fenêtre est fils de l’agent source.

Modèle PAC-Amodeus [nigay, 1991]

règle 9 :
Si la spécification d'une commande implique des actions distribuées sur plusieurs agents, ceux-ci doivent être placés sous le contrôle d'un agent qui cimente les actions réparties en une commande.



Modèle PAC-Amodeus [nigay, 1991]

règle 10 :
Un agent PAC et son fils unique peuvent être regroupés en un seul agent.

règle 11 :
Un agent dont le rôle peut être encapsulé par un objet de présentation ou d'interaction peut être éliminé et apparaître comme un composant de la présentation de son agent père.