

***Méthode TOOD***  
***pour la spécification et la conception des***  
***systèmes interactifs***

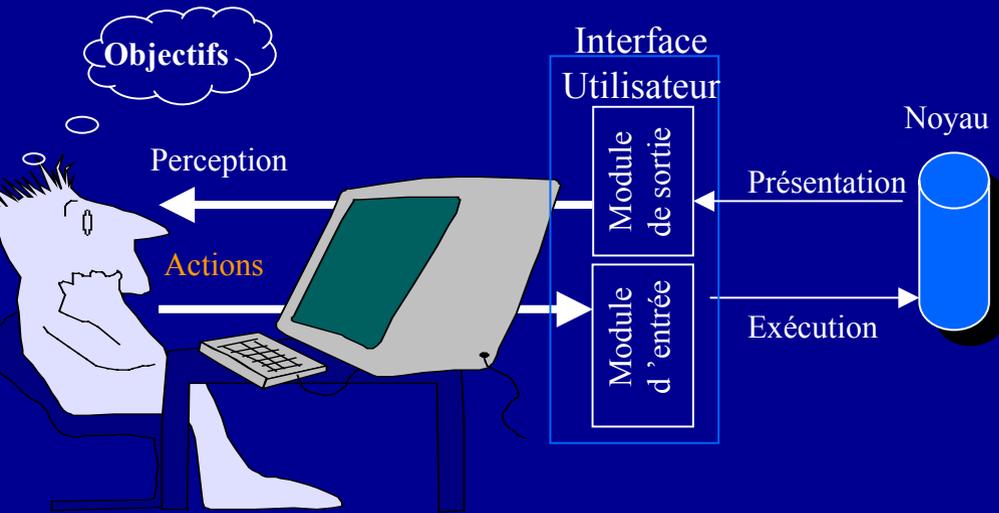
**Mourad ABED**

**Vendredi 07 Mai 2004**

**Réunion MFI à Paris**



# Introduction générale



- ◆ Percée des NTIC, prépondérance des systèmes interactifs dans tous les domaines (systèmes d'information, systèmes grand public, supervision...)

Nécessité de maîtrise de la conception des systèmes interactifs

⇒ Conception d'interface utilisateur basée sur les modèles (Model-Based user interface Design, MBD)



# *Plan*

---

- 1. Cycles de développement et outils de base en conception des IHM**
  - 2. TOOD : Modèle générique et formalismes de base**
  - 3. TOOD : Démarche méthodologique**
  - 4. TOOD : Environnement de développement**
- C&P Conclusion et perspectives de recherche**



# *Cycles de développement*

## *Modèles classiques du génie logiciel :*

- **modèle en cascade** [Royce, 1970], [Boehm, 1981]
- **modèle en V** [McDermid et Ripkin, 1984]
- **modèle spirale** [Boehm et al., 1984]
- **modèle en X** [Verilog, 1991]
- ...

## *Limites vis-à-vis des systèmes interactifs :*

- **trop génériques**
- **analyse de tâche**
- **facteurs humains**
- **prototypage**
- **évaluation itérative**



# *Cycles de développement*

---

*Modèles enrichis pour le développement des systèmes interactifs* [Long et Denley, 1990] [Curtis et Hefley, 1994] :

- **modèle en Étoile** [Hix et al., 1993]
- **modèle en V étendu** [Coutaz, 95]
- **modèle Nabla** [Kolski, 97]
- ...

*Apports :*

- **prise en compte des facteurs humains**
- **cycles itératifs**

*Limites :*

- **manque de couverture des étapes**
- **connaissance et acceptation par le concepteur**

# Outils de base : Modèles de la Tâche

## Cycle en V

### Modèles de la Tâche (MT)

ALG [Reisner, 1984]

CLG [Moran, 1981],

TAG [Payne et Green

ETAG [Tauber

JAN [J

HTA

GO

KLM

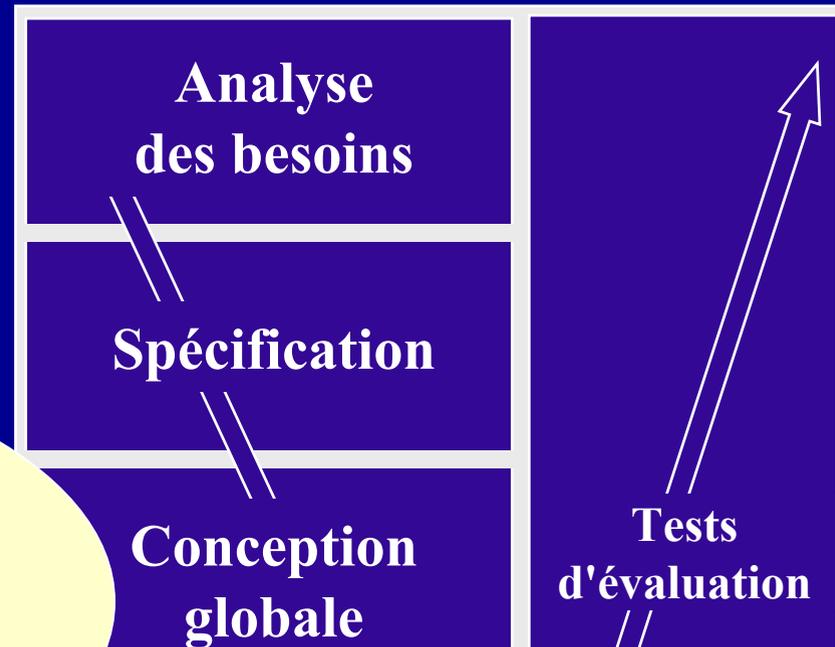
TKS [J

KB [Boy, 1989]

- planification de tâches
- relations entre tâches
- représentation des états du système et des séquences opératoires

#### Limites :

- sémantique informelle
- représentation inexistante ou partielle des données utilisées par les tâches
- peu adaptés aux contextes multi-utilisateur
- faible prise en compte des interruptions





# Outils de base : Modèles de l'Utilisateur

## Cycle en V

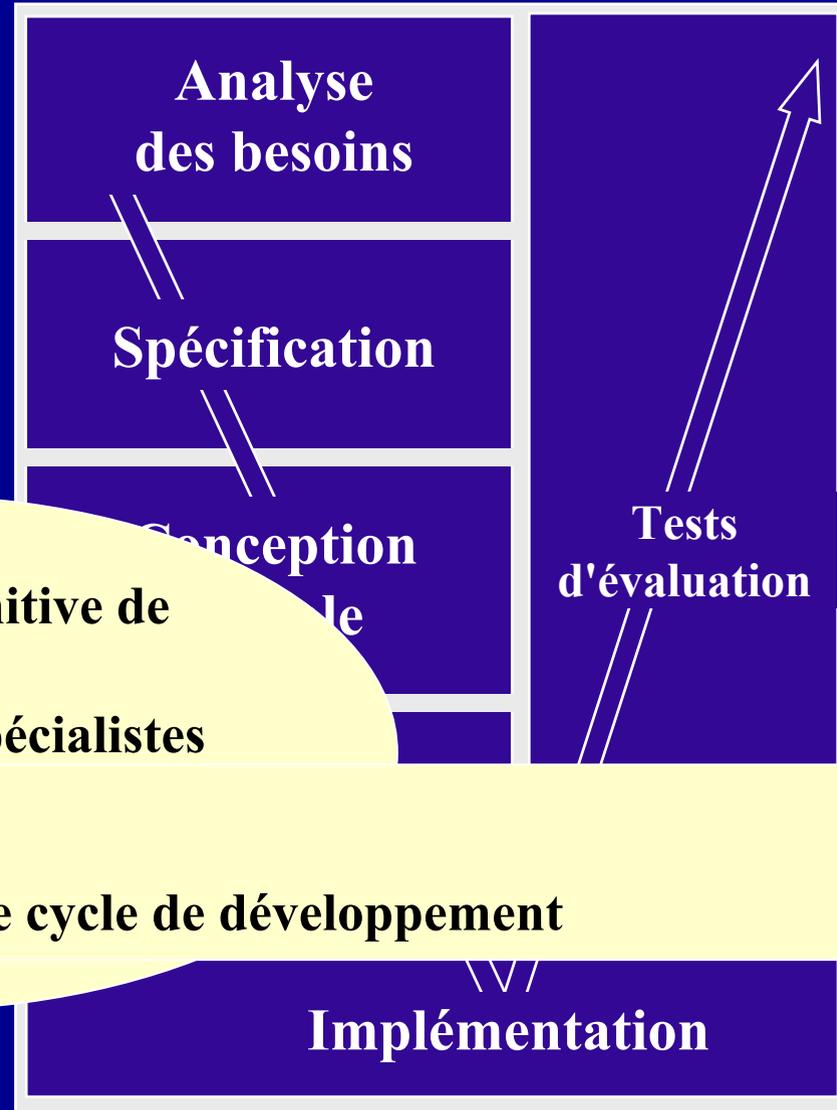
### Modèles de l'Utilisateur (MU)

ACT\* [Anderson, 1983],  
CS [Barnard et May, 1994],  
CCT [Kieras et Polson, 1985],  
COSIMO [Cacciabue et al., 1992],  
Théorie de l'action [Norman, 1986],  
Modèle de Rasmussen [Rasmussen, 1986]

- Modélisation cognitive de l'utilisateur
- Intervention de spécialistes

#### *Limite :*

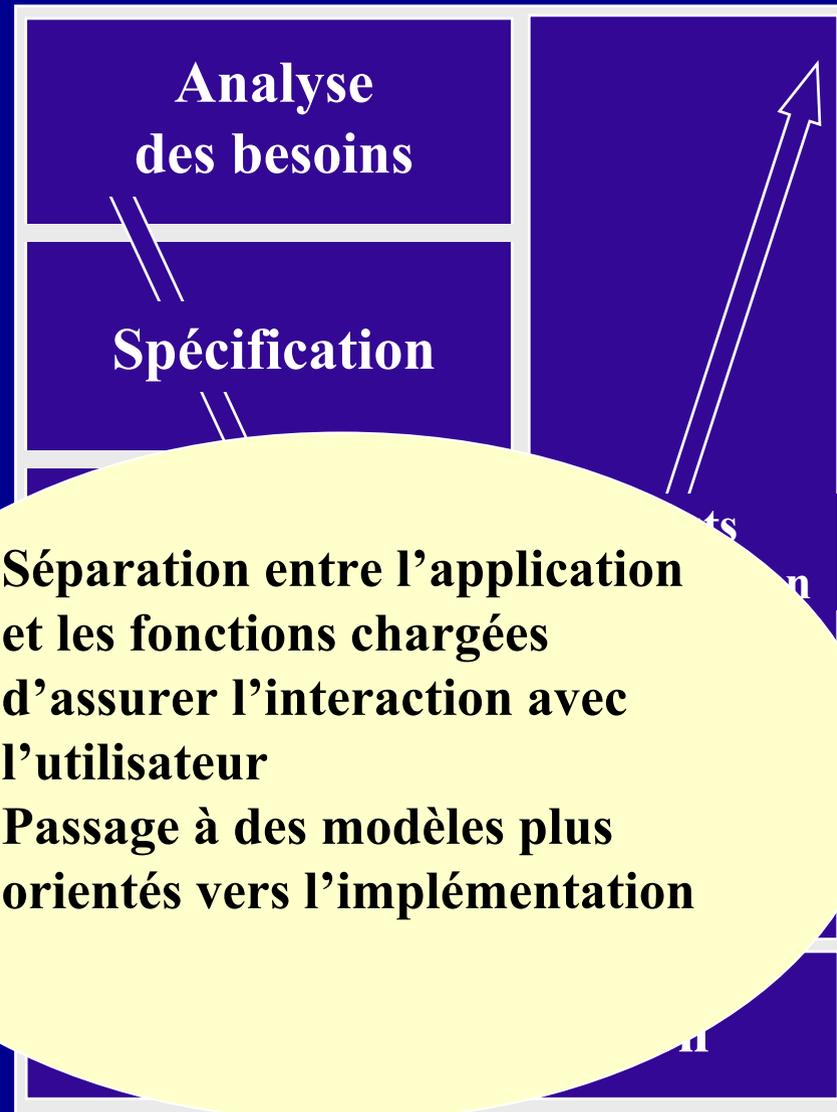
- Peu intégrés dans le cycle de développement





# Outils de base : Architectures

## Cycle en V



## Architectures Conceptuelles

**SEEHEIM** [Pfaff, 1985],  
**ARCH** [Bass et al., 1991],  
**PAC** [Coutaz, 1987],  
**Pac-Amodeus** [Nigay, 1994]

..

- Séparation entre l'application et les fonctions chargées d'assurer l'interaction avec l'utilisateur
- Passage à des modèles plus orientés vers l'implémentation



# Outils de base : Formalismes

## Cycle en V



## Formalismes de conception

**HOOD** [Michel, 1991],

**OOA, OOD** [Coad et Yourdon, 1991],

**OMT** [Rumbaugh et al., 1991],

**UML** [Booch et al., 1999],

**Formalismes à états** [Woods, 1991],

**RdP et ses variantes** [Petri, 1977],

[Jensen, 1980], [David et Alla, 1992]

**ICO** [Palanque et Bastide, 1995]

...

**Choix d'un formalisme de spécification**

=

**décision stratégique du processus de développement**

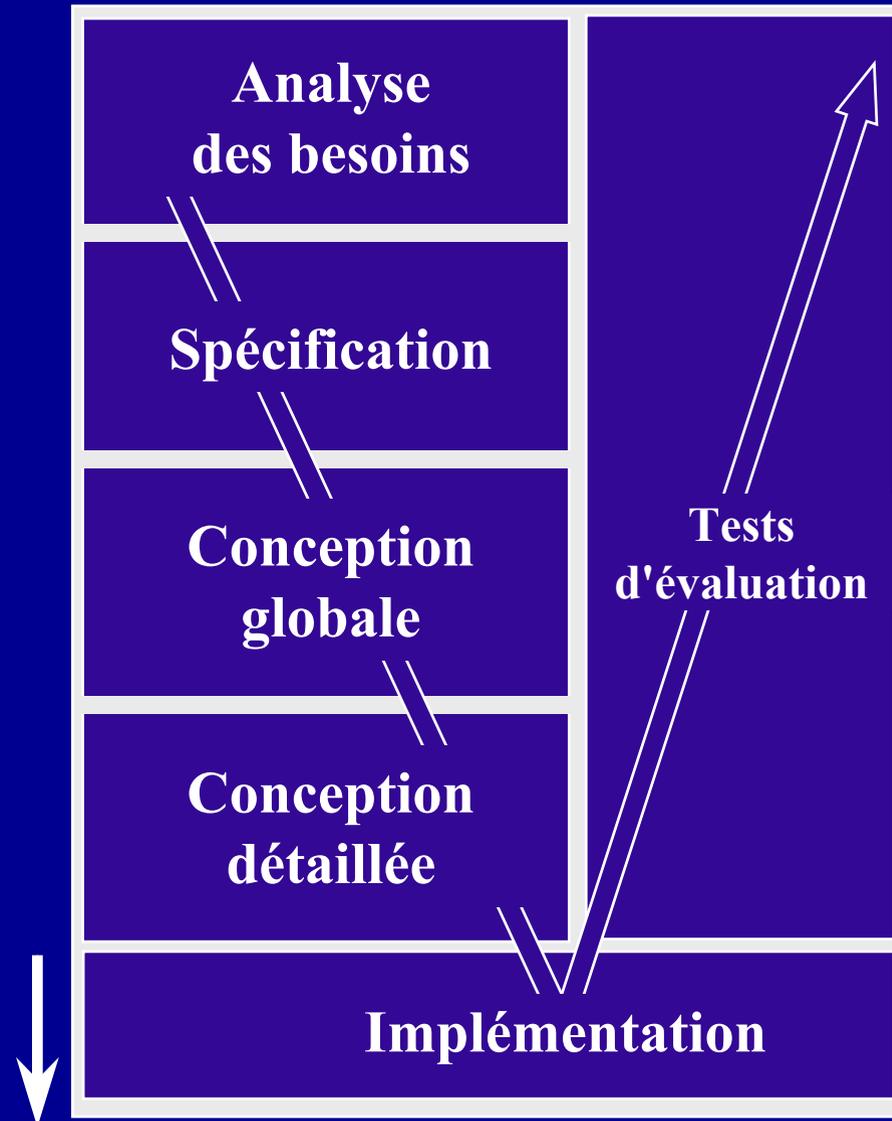
**Critères :**

**pouvoir d'expression, adéquation avec le cycle de développement, aspects (temporels, concurrence, interruptions...), analyse prédictive (vérification de propriétés du système, performance...), capacité de générer tout ou partie du système...**



# Outils de base : Implémentation

## Cycle en V



## Environnements et outils d'implémentation

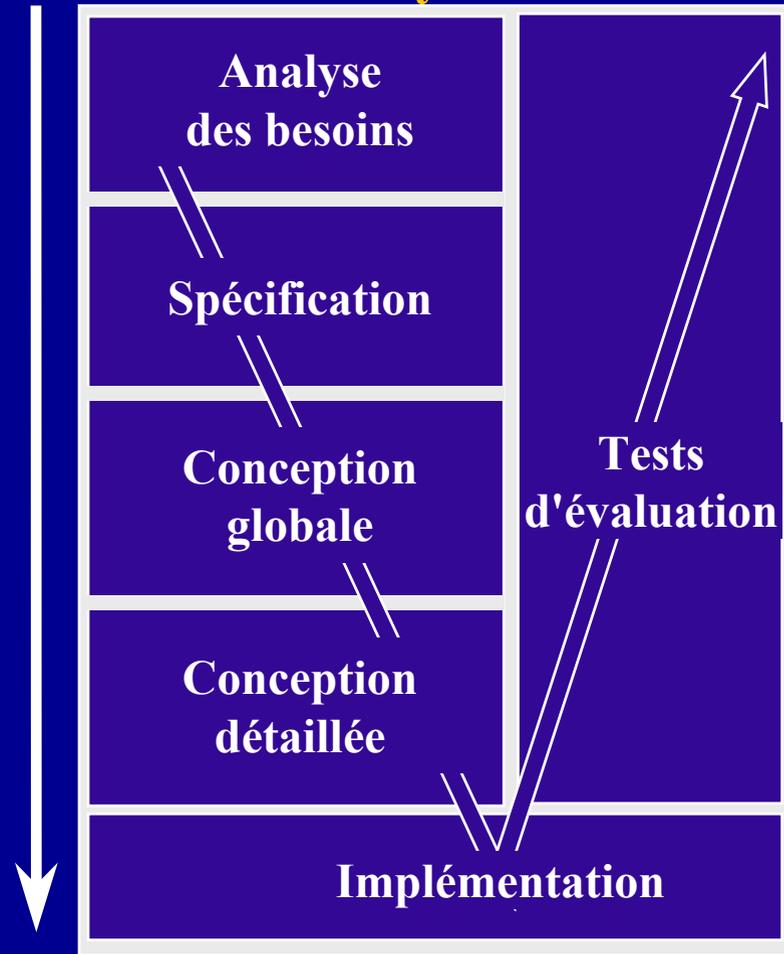
**ILOG** [ILOG, 1994],  
**OpenStep** [Next, 1992],  
**Java AWT** [Geary et Mc Clellan, 1997],  
**Java Swing** [Geary, 1999]

...



# Conception à base de modèles

## Cycle en V



## Approche à base de modèles

**MASTERMIND** [Szekely et al., 1995],

**MOBI-D** [Puerta et Maulsby, 1997],

**ADEPT** [Johnson et al., 1993],

**TADEUS** [Elwert et al., 1994],

**ALACIE** [Gamboa-Rodriguez, 1997],

**DIANE+** [Tarby, 1993],

**TRIDENT** [Bodart et al., 1995],

[Vanderdonckt, 1997],

**FUSE** [Lonzewski et Schreiber, 1996],

**TLIM** [Paternò et Mancini, 1999],

**CTT/ICO** [Navarre, 2001],

...

CADUI ; CHI ; UIST ; IHM



## **Modèle**

- **Modèle de la Tâche,**
- **Modèle du Domaine,**
- **Modèle de l'Utilisateur**

**Modèle Abstrait de  
l'Interface**

**Modèle Concret de  
l'Interface**

- ◆ **Modèle de la tâche (MT)**
  - **Représente la tâche utilisateur**
  - **Ou la tâche informatisée**
- ◆ **Modèle du domaine (MD)**
  - **Objets manipulés par l'application interactive**
  - **Utilisation dans les tâches**
- ◆ **Modèle de l'utilisateur (MU)**
  - **Liste de préférences**
  - **Représentation détaillée du comportement de l'utilisateur**



# MB-IDE : Conclusion

Critères MB-IDE	Modèles					Nombre de Notation(s)
	Tâche	Domaine	Utilisateur	Abstrait	Concret	
MASTERMIND	X	X			X	1
MOBI-D	X		X		X	1
ADEPT	X		X	X	X	plusieurs
TADEUS	X	X	X	X	X	plusieurs
ALACIE	X			X	X	plusieurs
DIANE+	X	X			X	plusieurs
TRIDENT	X	X		X	X	plusieurs
FUSE	X	X	X	X	X	plusieurs
TLIM	X			X	X	plusieurs
CTT/ICO	X	X		X	X	plusieurs



# MB-IDE : Conclusion

Critères MB-IDE	Modèles					Nombre de Notation(s)
	Tâche	Domaine	Utilisateur	Abstrait	Concret	
MASTERMIND	X	X			X	1
MOBI-D	X		X		X	1
ADEPT	X		X	X	X	plusieurs
TADEUS	X	X	X	X	X	plusieurs
ALACIE	X			X	X	plusieurs
DIANE+	X	X			X	plusieurs
TRIDENT	X	X		X	X	plusieurs
FUSE	X	X	X	X	X	plusieurs
TLIM	X			X	X	plusieurs
CTT/ICO	X	X		X	X	plusieurs





# *Conclusion*

---

## ◆ **Exigences**

- **Représentation de la tâche utilisateur**
- **Intégration d'un modèle de l'utilisateur**
- **Considération d'une architecture**
- **Utilisation de formalismes utilisables tout au long du cycle**



# Plan

---

1. **Cycles de développement et outils de base en conception des IHM**
  2. **TOOD : Modèle générique et formalismes de base**
  3. **TOOD : Démarche méthodologique**
  4. **TOOD : Environnement de développement**
- C&P Conclusion et perspectives de recherche**

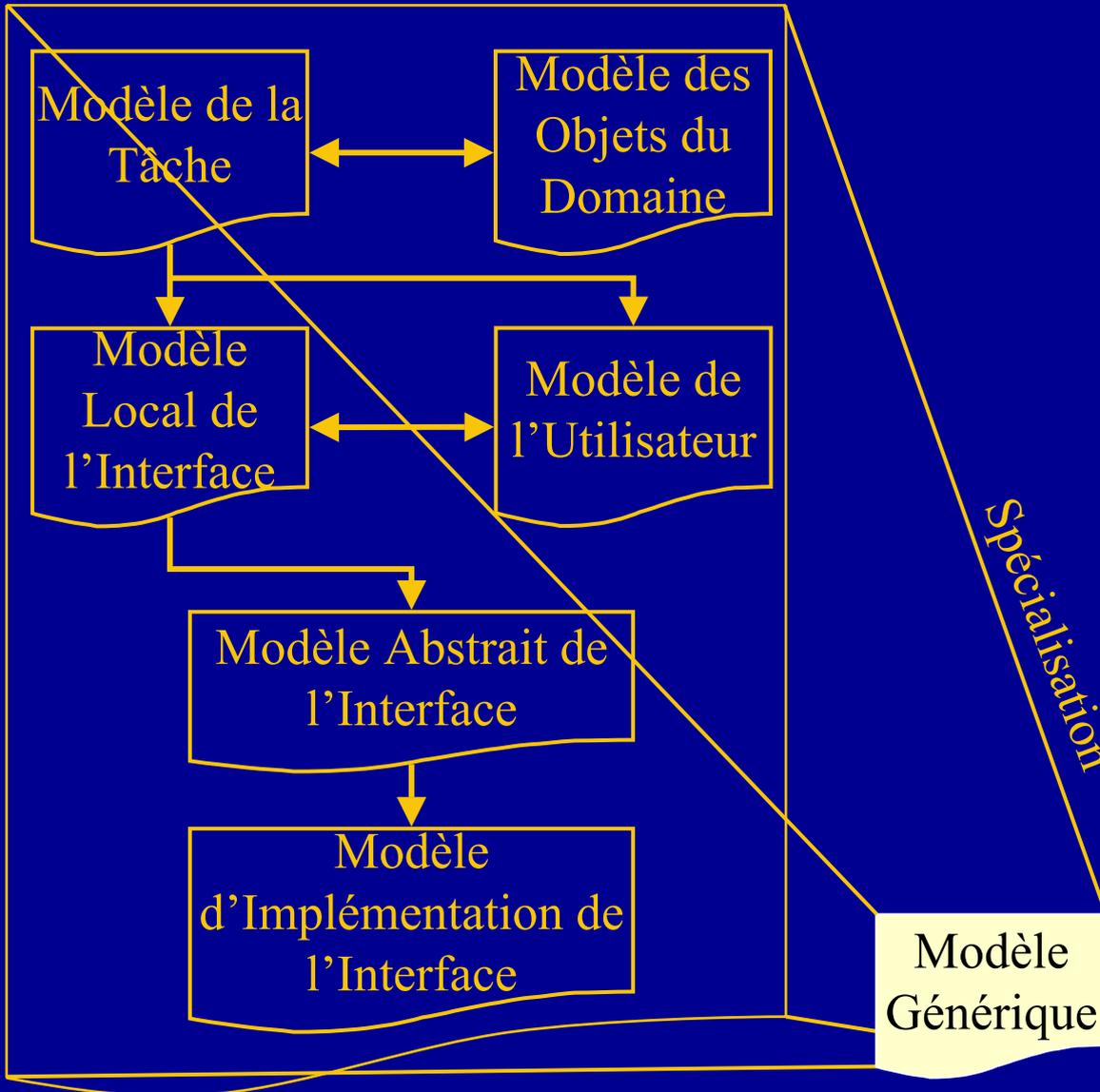


# *Objectifs*

---

- ◆ **Fournir un support méthodologique pour l'analyse de la tâche et la conception d'interface**
- ◆ **Utiliser un formalisme unique**
- ◆ **Modéliser les contextes multi-utilisateurs**
- ◆ **Fournir un outil de capitalisation de conception**

# TOOD : modèle générique





# ***TOOD : modèle générique***

---

**? Prise en compte simultanée des aspects statiques et des aspects dynamiques**

**↳ Orientation Objet**

**(Aspects de structuration :  
classification, encapsulation, héritage...)**

**&**

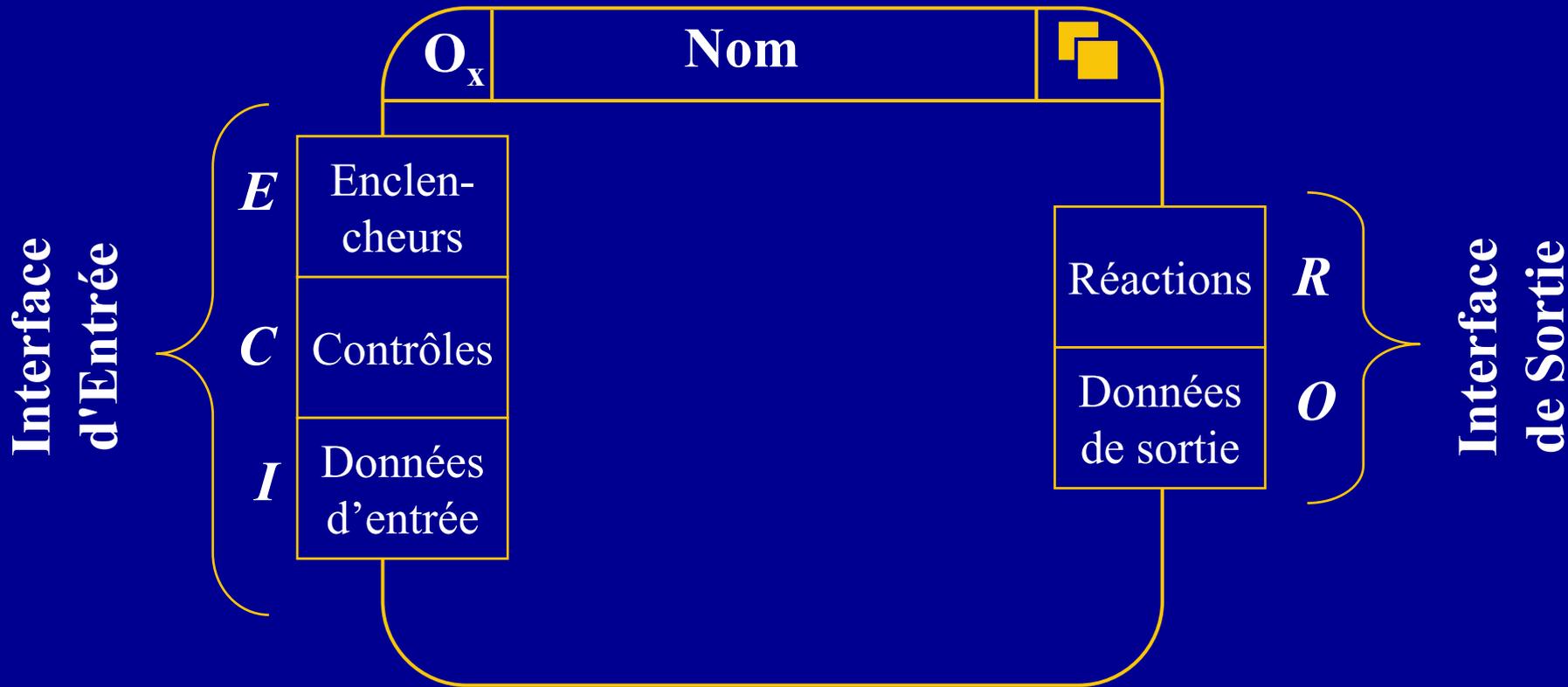
**↳ Réseaux de Petri Objet**

**(Aspects dynamiques ou comportementaux :  
concurrence, synchronisation...)**



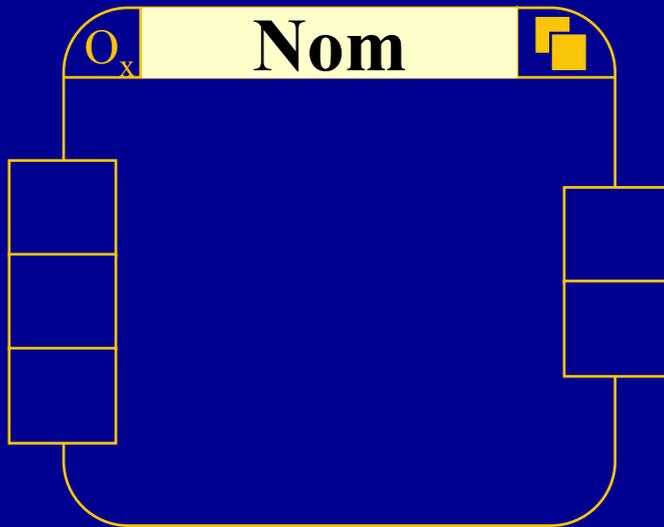
# Structure de la Classe TOOD

**Classe\_TOOD** = {Interface d'entrée, Interface de sortie, Identificateurs}





# Identificateurs de la Classe-TOOD



## Nom

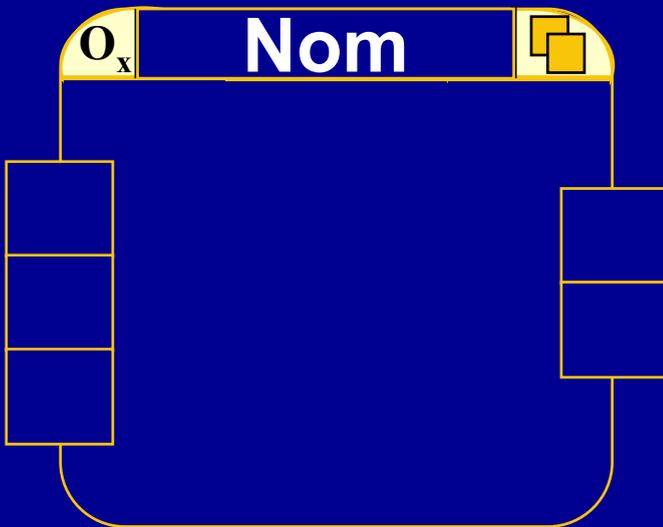
- Définit l'objet dans le vocabulaire utilisé par les utilisateurs

## Description

- Explication dans les termes usuels du domaine



# Identificateurs de la Classe-TOOD



## Indice

- identifiant formel de l'objet

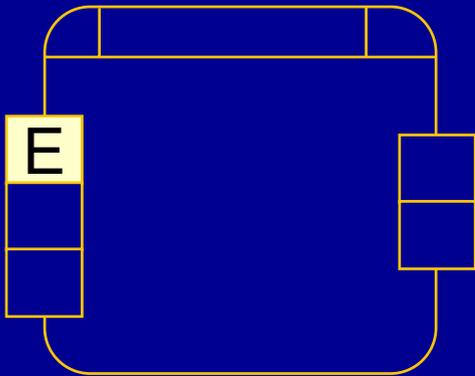
## Hiérarchie

- Organisation de composition
- Les carrés représentent le nombre d'objets fils composant l'objet composé



# *Descripteurs de la Classe-TOOD*

---

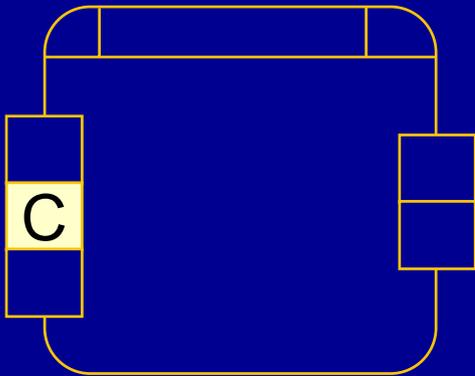


**Enclencheurs** : Evénements qui déclenchent la réalisation d'un service par l'objet



# *Descripteurs de la Classe-TOOD*

---

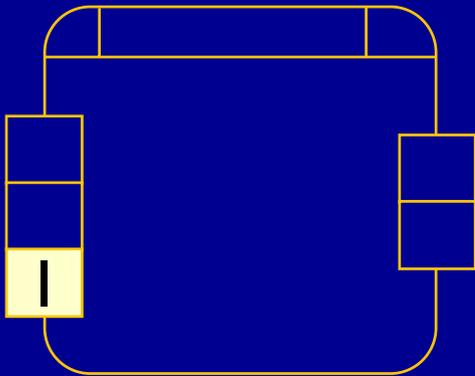


**Contrôles** : Informations qui doivent être vérifiées durant un service, données non modifiables dans la réalisation d'un service



# *Descripteurs de la Classe-TOOD*

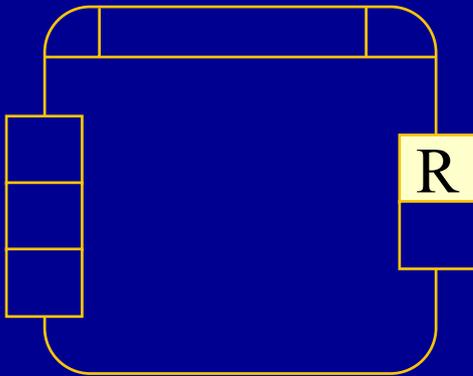
---



**Données d'entrée** : Informations demandées lors de l'exécution du service, modifiables pendant celle-ci



# Descripteurs de la Classe-TOOD

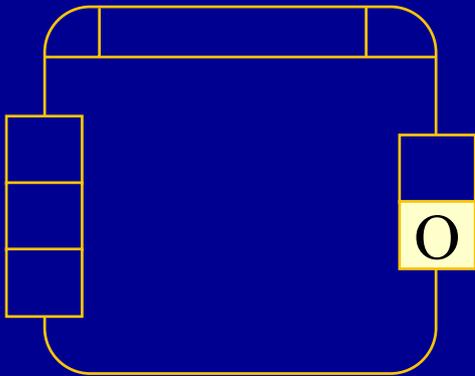


**Réactions** : Résultats produits par la réalisation du service. En fonction de cette réaction, répétition du service ou déclenchement d'un autre



# *Descripteurs de la Classe-TOOD*

---

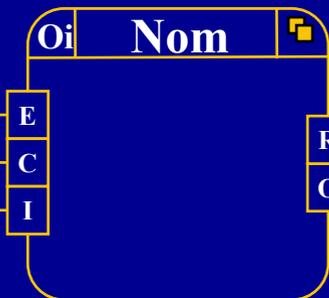
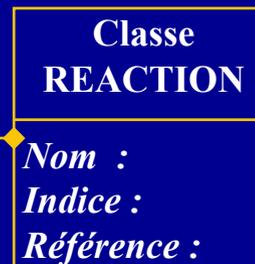


**Données de sortie :** Données transformées ou créées par le service



# Objets-descripteurs

Demande du service 1



ok





# *Modèle dynamique*

---

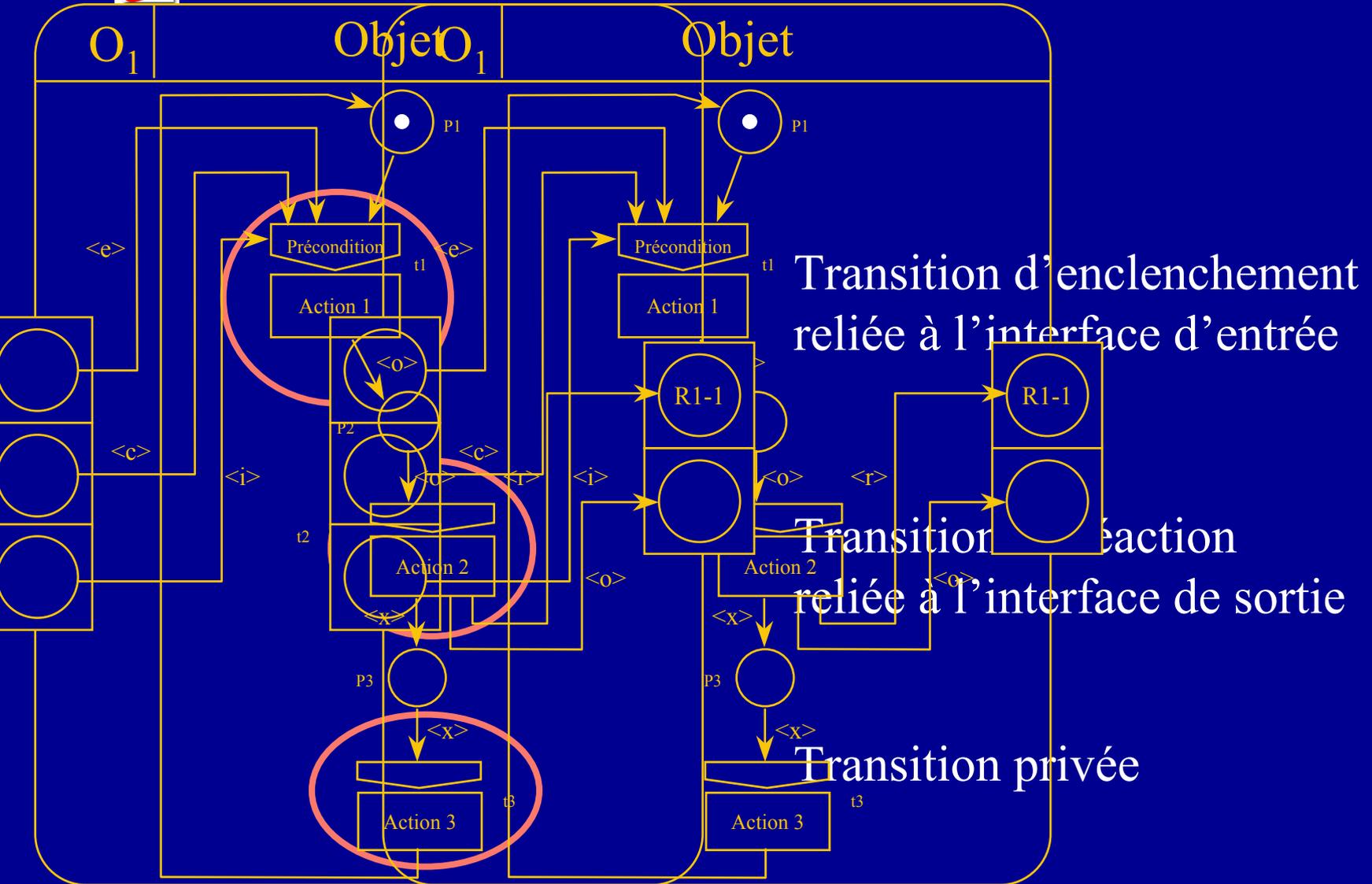
- ◆ **Décrire le comportement de l'objet**
- ◆ **Préciser les coopérations inter-objets**
- ◆ **Définir la structure d'exécution des actions lors d'un appel de service**



**Utilisation des Réseaux de Petri Objet  
(RPO) [Sibertin-blanc, 1985]**

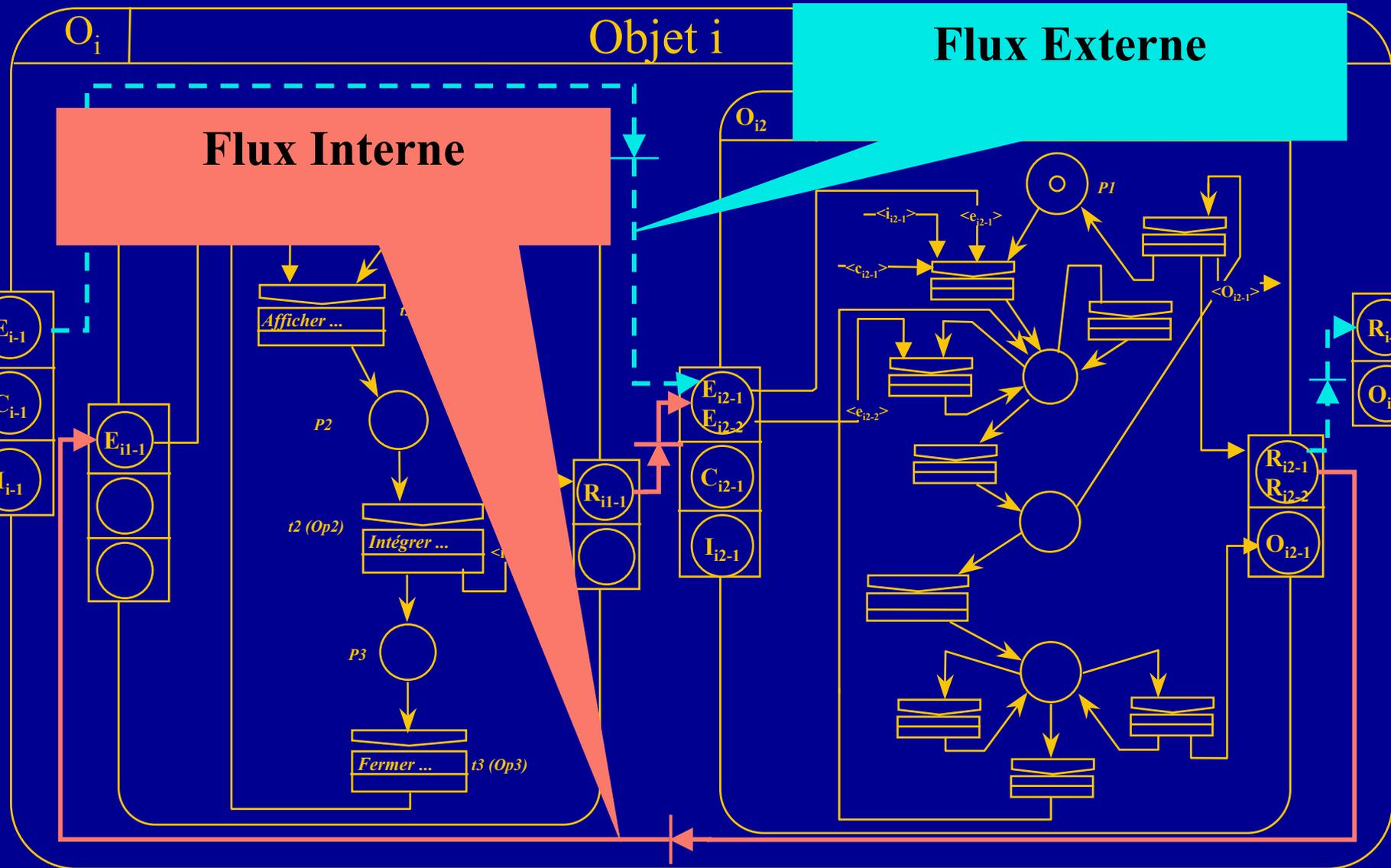


# Modèle dynamique : ObCS

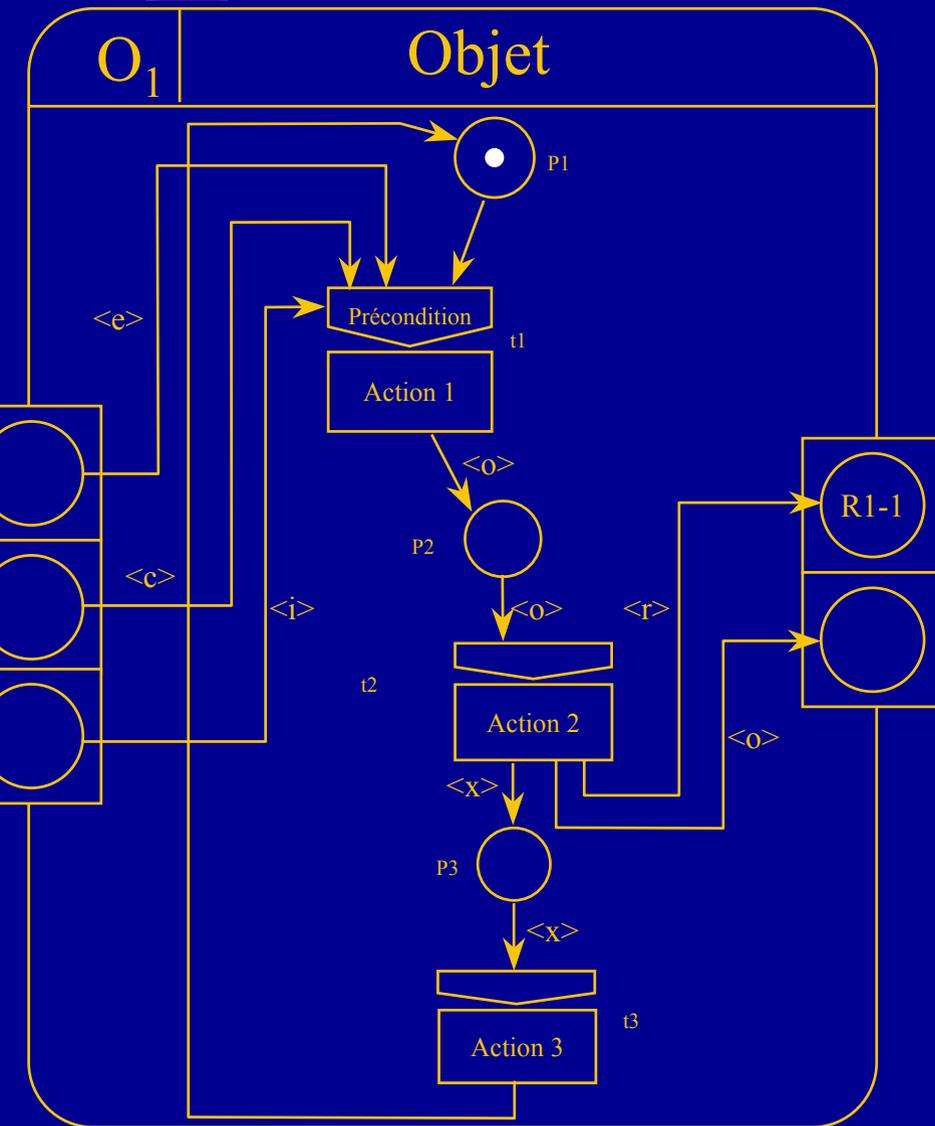




# Communications inter-objets



# TOOD : modèle générique



## Classe <Nom>

### Identification

Indice :  $O_i$

Description :

Est un :

Objet\_père :

Objets\_fils :

### Spécification

Événements :

- E1-1 :--

-

Contrôle/Commande:

- C1-1 :

Entrées:

- I1-1 :

Sorties:

- O1-1

Réactions:

- R1-1 :--

-

### Implémentation

Attributs:

- att1 :--

- att2 :--

Actions:

- t1 :DECLENCHEMENT :--

Param : < E1-1; C1-1; I1-1 >;

Précon : C1-1 = vrai

Act : Att1 := 1; Att2 := I1-1

//Description du service... //

...



# Plan

---

1. **Cycles de développement et outils de base en conception des IHM**
  2. **TOOD : Modèle générique et formalismes de base**
  3. **TOOD : Démarche méthodologique**
  4. **TOOD : Environnement de développement**
- C&P Conclusion et perspectives de recherche**

# Cycle de développement de *TOOD*



Analyse des besoins

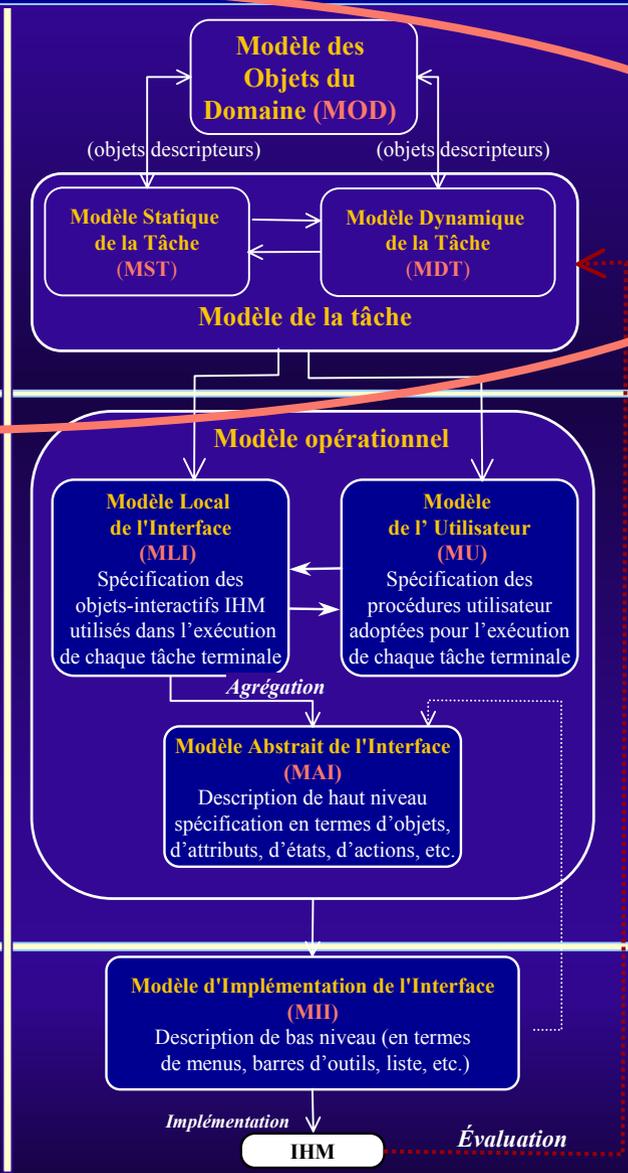
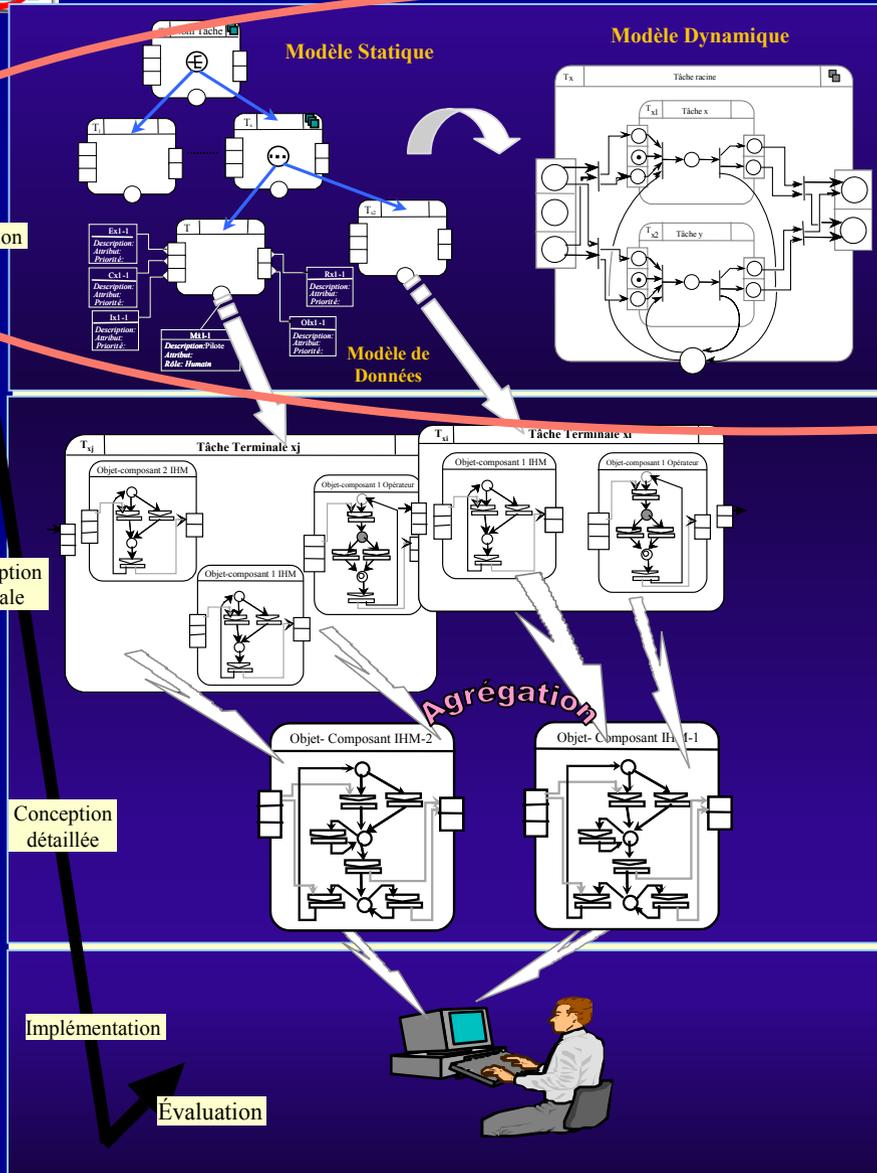
Spécification

Conception globale

Conception détaillée

Implémentation

Évaluation



# Cycle de développement de TOOD

Analyse des besoins

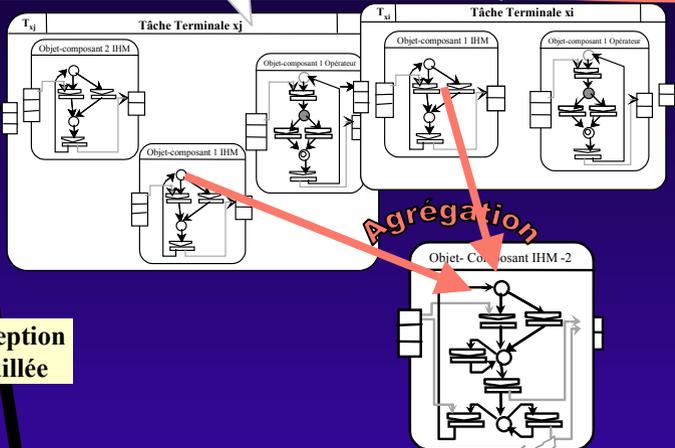
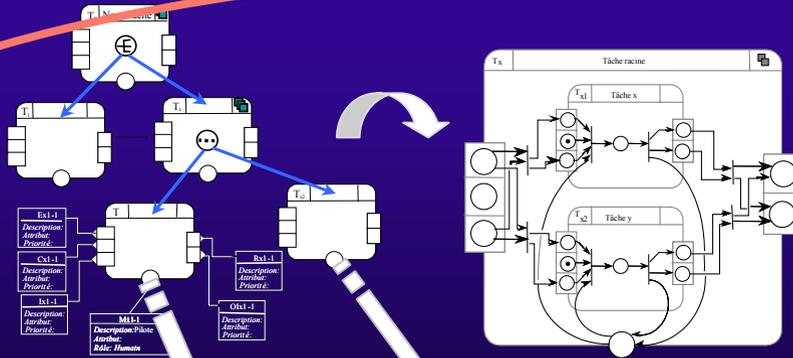
Spécification

Conception globale

Conception détaillée

Implémentation

Évaluation



Modèle des Objets du Domaine (MOD)

Modèle de la tâche

Modèle Statique de la Tâche (MST)

Modèle Dynamique de la Tâche (MDT)

Modèle opérationnel

Modèle Local de l'Interface (MLI)

Modèle de l'Utilisateur (MU)

Agrégation  
Modèle Abstrait de l'Interface (MAI)

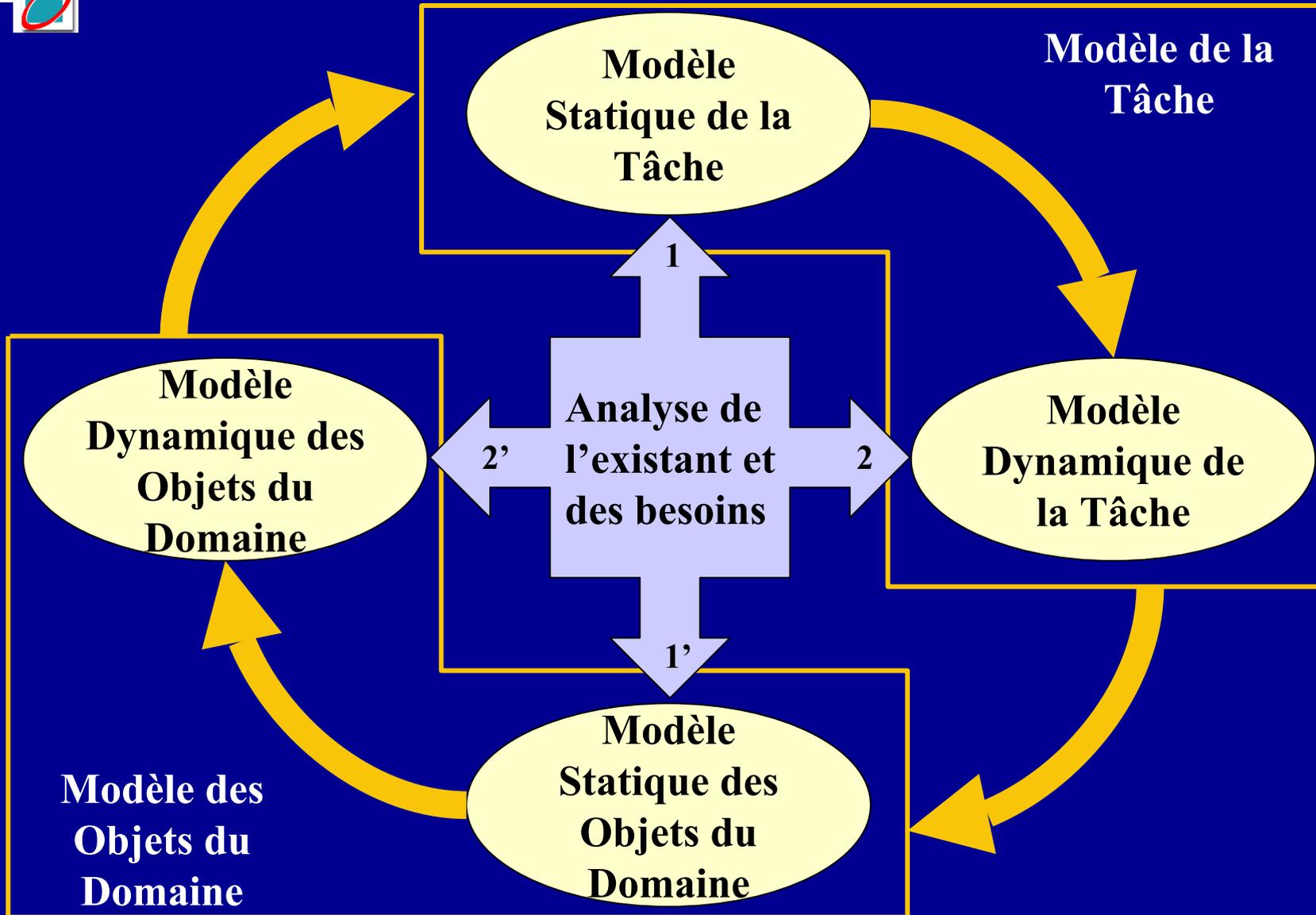
Modèle d'Implémentation de l'Interface (MII)

Implémentation

IHM

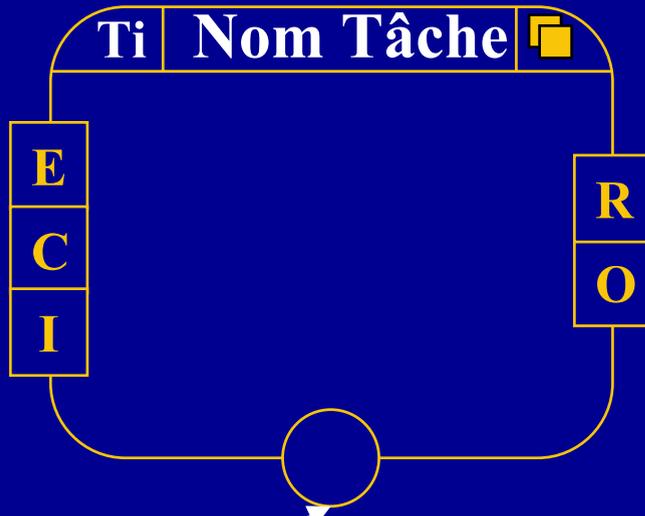
Évaluation

# Modèle des objets du domaine



# Modèle statique : Classe\_Tâche

Interface  
d'Entrée (IE)



Interface de  
Sortie (IS)

**Ressources**

Opérateurs humains et/ou  
composants de l'application

Types de tâche	Ressources Humaines	Ressources Système
Manuelle	1..1	0
Automatique	0	1..N
Interactive	1..1	1..N
Coopérative	2..N	0..N

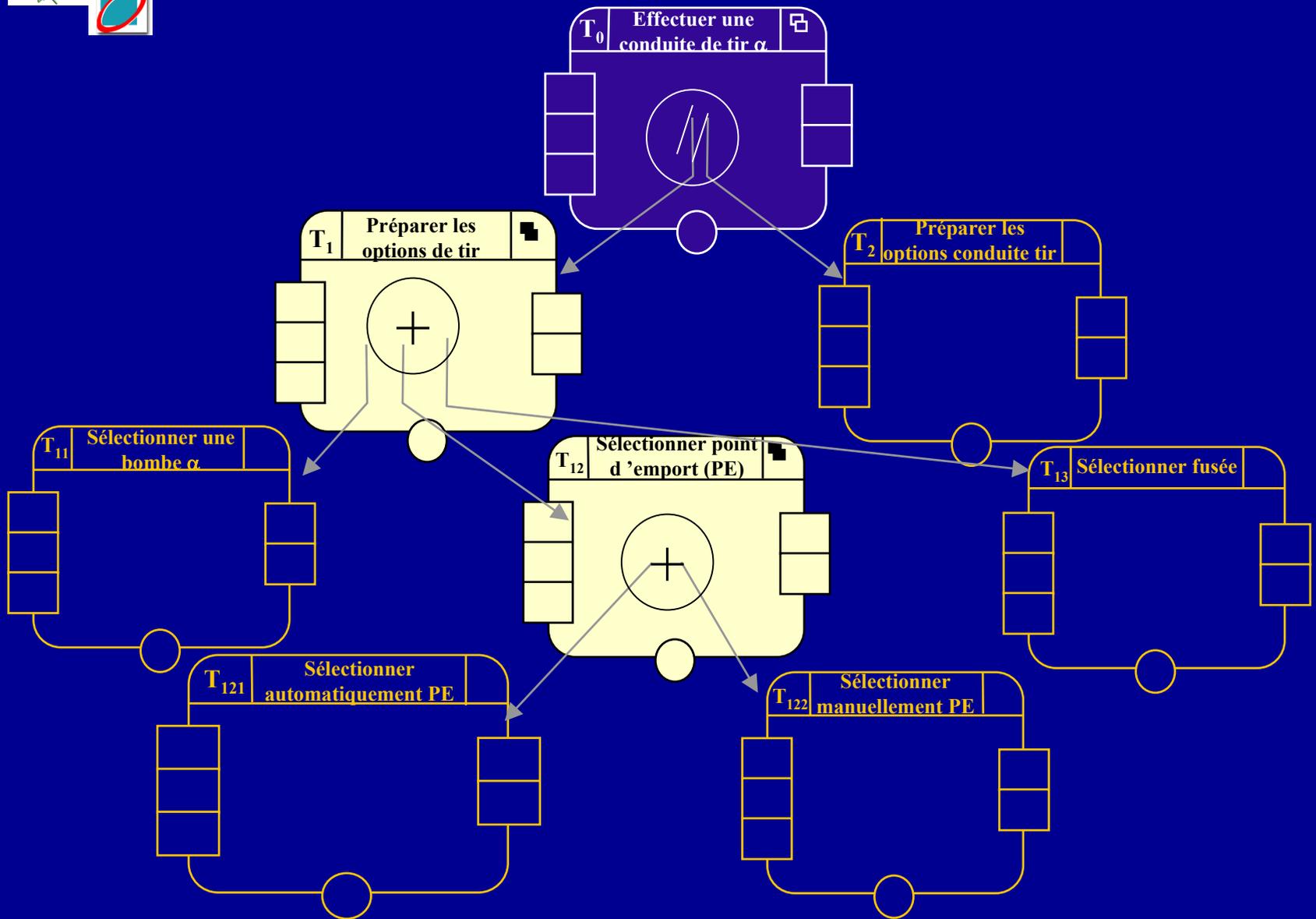


# *Construction du modèle de la tâche*

---

- 1. Identifier la classe-tâche globale**
- 2. Spécifier les objets descripteurs**
- 3. Décomposer la classe-tâche en sous classes-tâches**
- 4. Identifier de façon informelle les relations inter-tâches**
- 5. Continuer le processus itératif jusqu'à l'identification de toutes les classes-tâches terminales**

# Décomposition hiérarchique





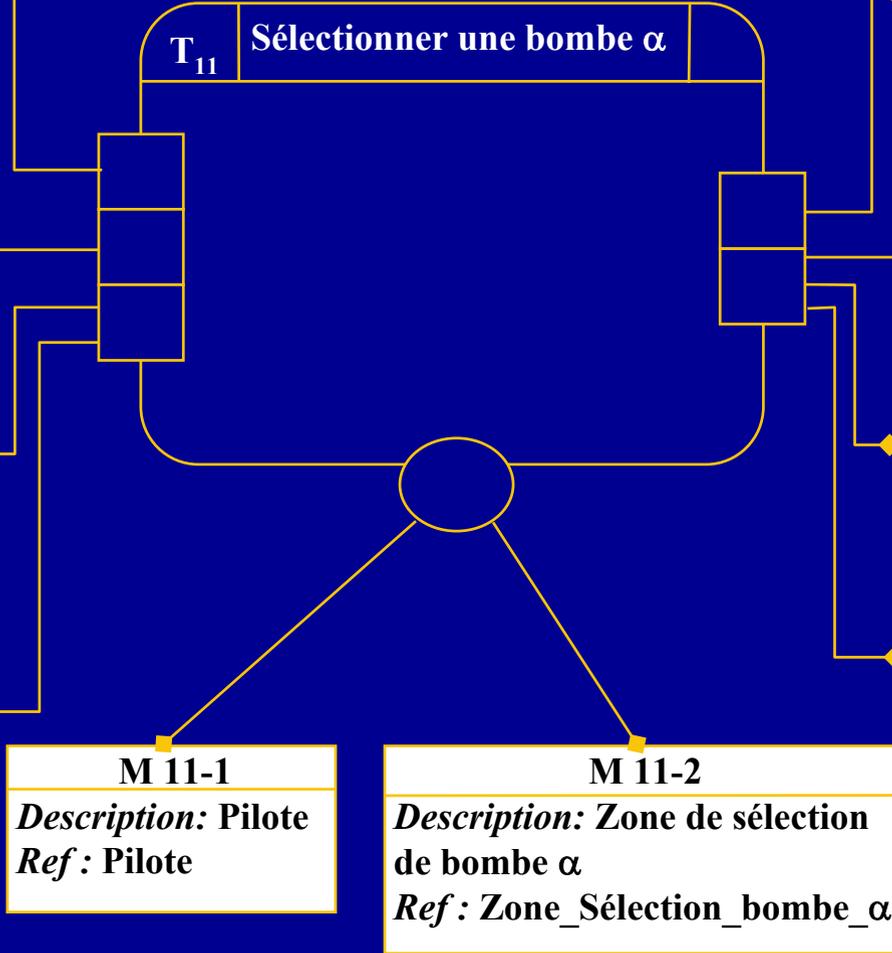
# Objets descripteurs de $T_{11}$

**E 11-1**  
*Description* : Evénement  
l'activation de la  
sélection d'une bombe  
*Ref* : Sélecteur  $\alpha$   
*Priorité*:1

**C 11-4**  
*Description*: Mode de  
sélection courant  
*Ref* : Mode.courant

**I 11-1**  
*Description*: Donnée  
représentant le capteur  $\beta$   
*Ref* : Capteur  $\beta$

**I 11-2**  
*Description*: Donnée  
représentant le capteur  $\tau$   
*Ref* : Capteur  $\tau$



**R 11-1**  
*Description*: Indique le  
résultat de sélection d'une  
bombe  
*Ref* : Bombe.active

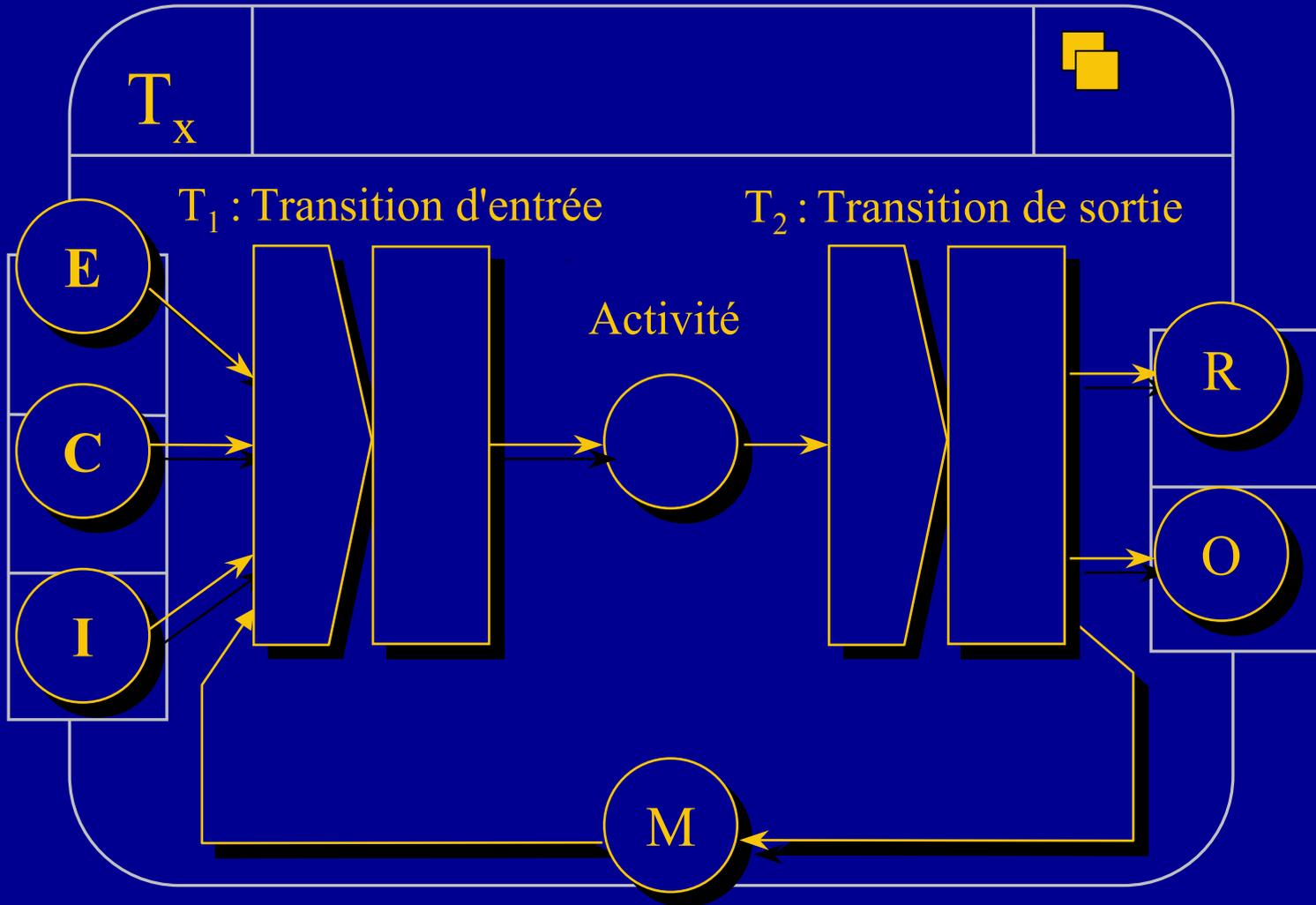
**O 11-1**  
*Description*: Bombe  
sélectionnée  
*Ref* : Bombe

**O 11-2**  
*Description*: Donnée  
représentant le capteur  $\beta$   
*Ref* : Capteur  $\beta$

**O 11-3**  
*Description*: Donnée  
représentant le capteur  $\tau$   
*Ref* : Capteur  $\tau$

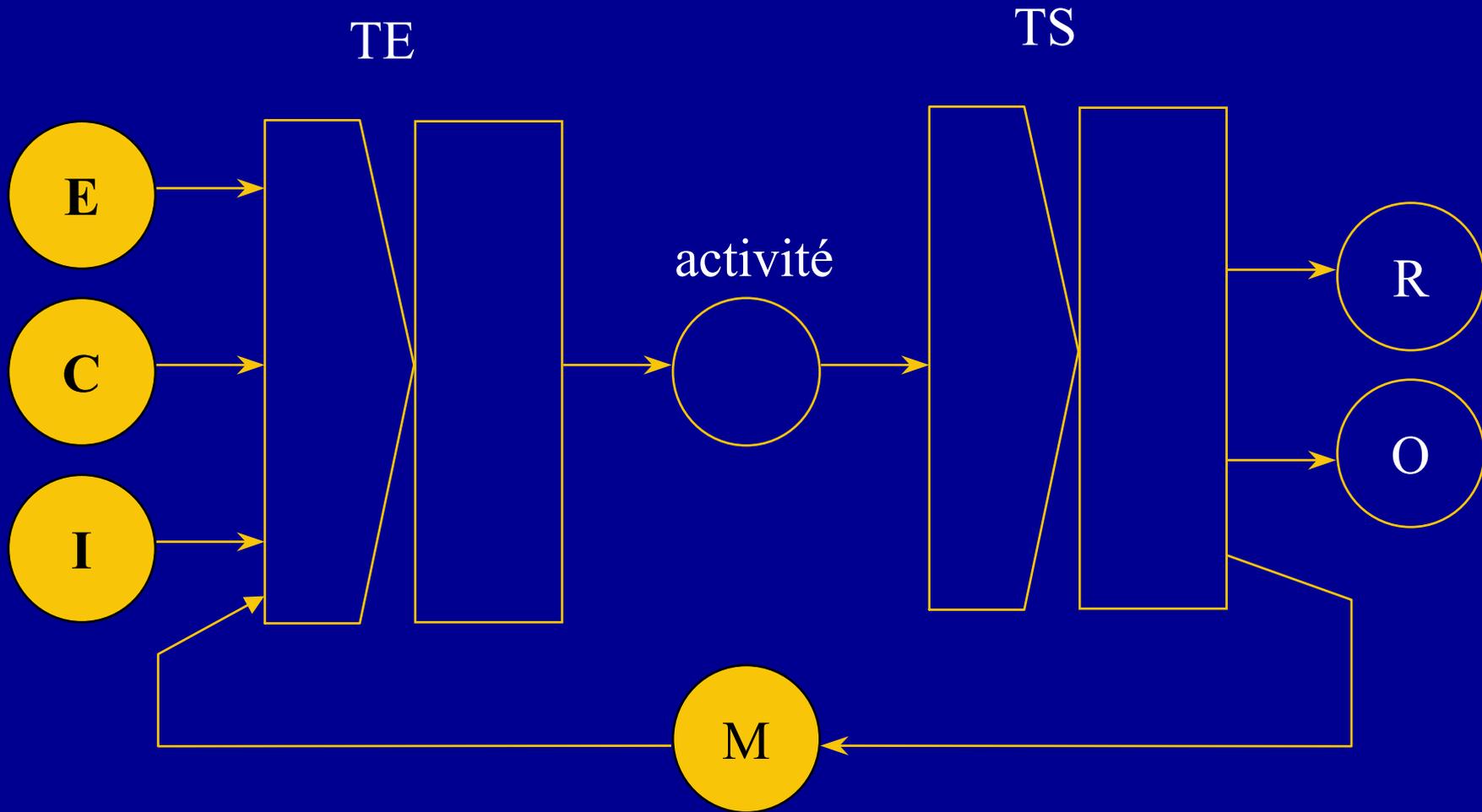


# TCS : Structure de Contrôle de la Tâche



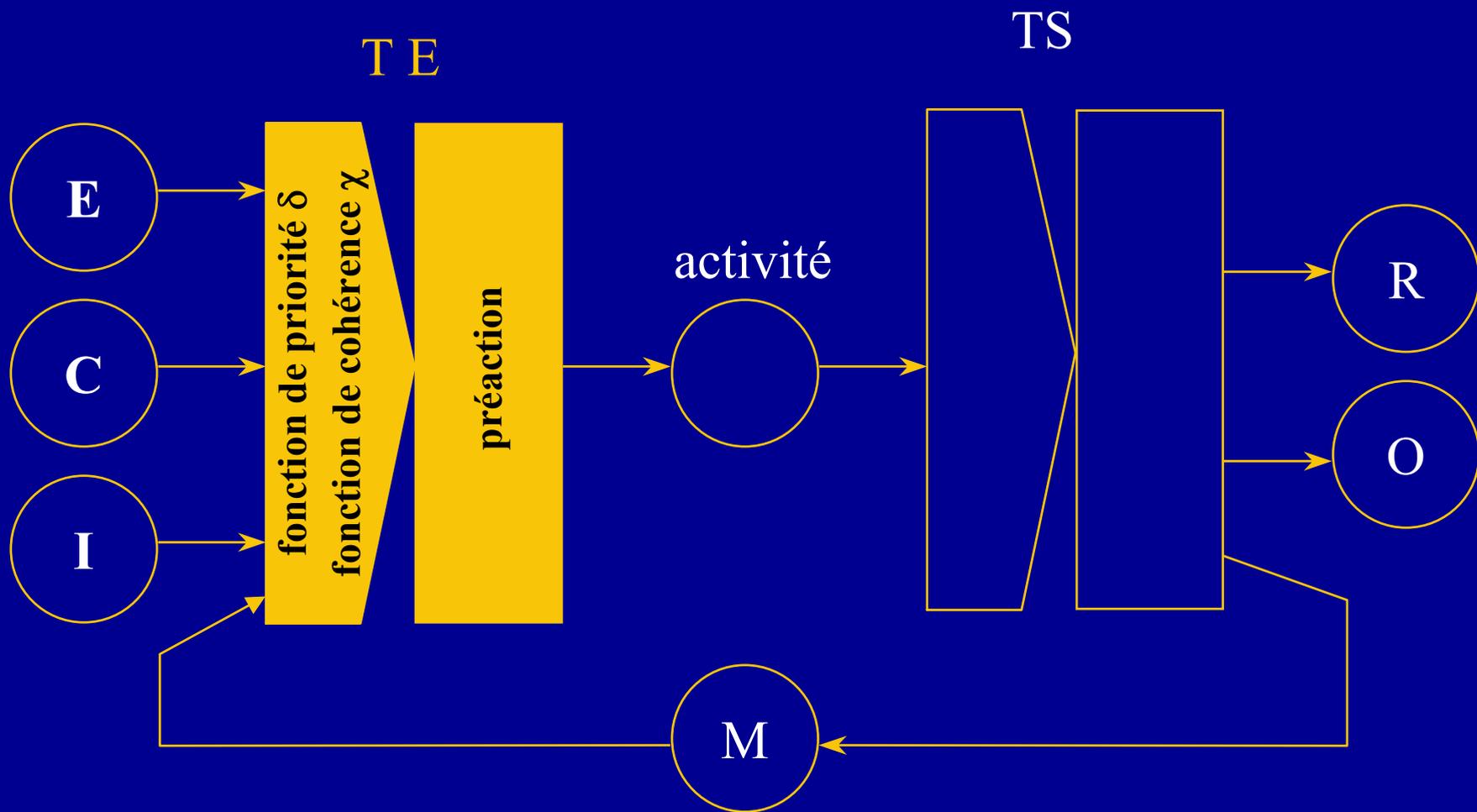


# TCS : Interface d'entrée



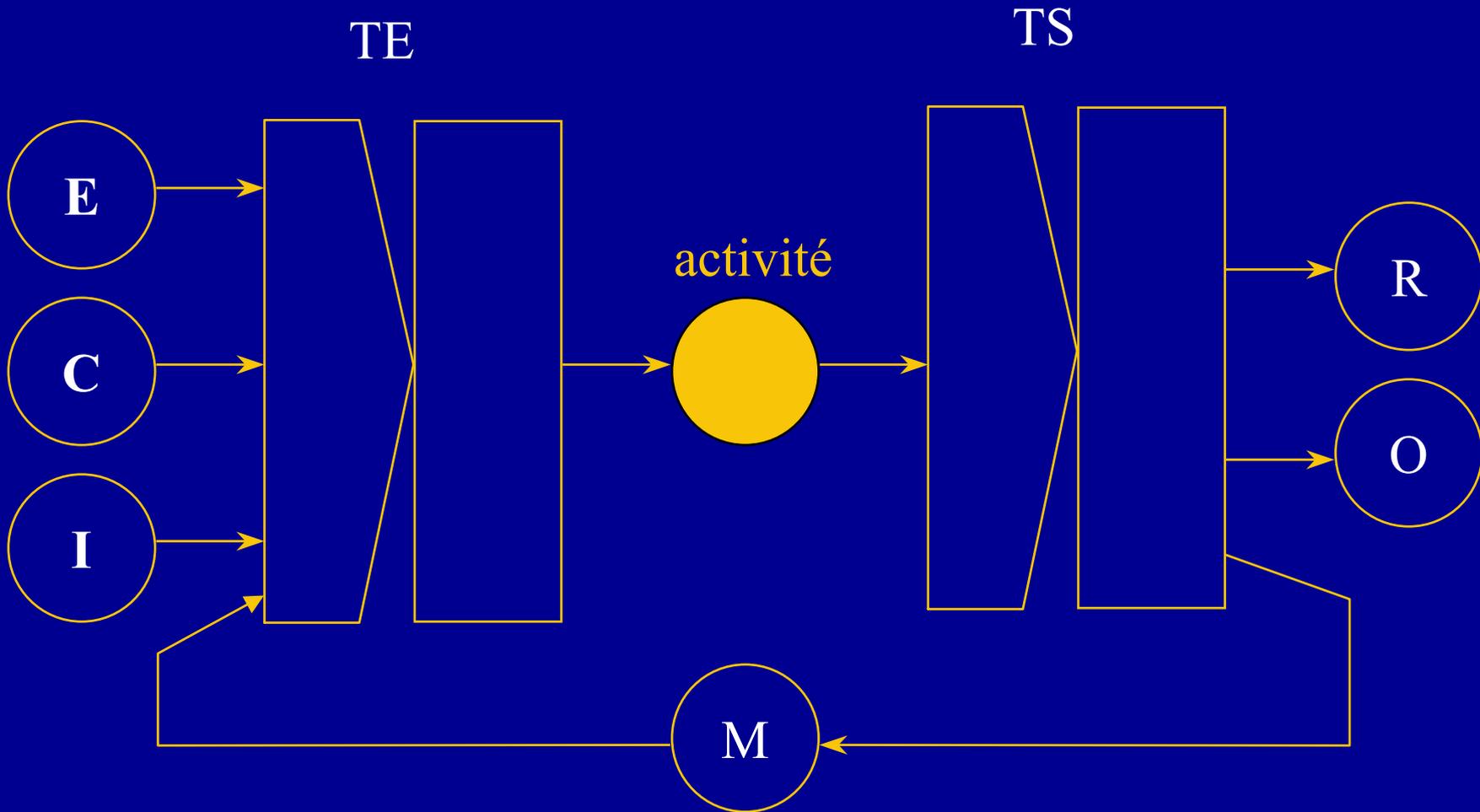


# *TCS : Transition d'entrée*



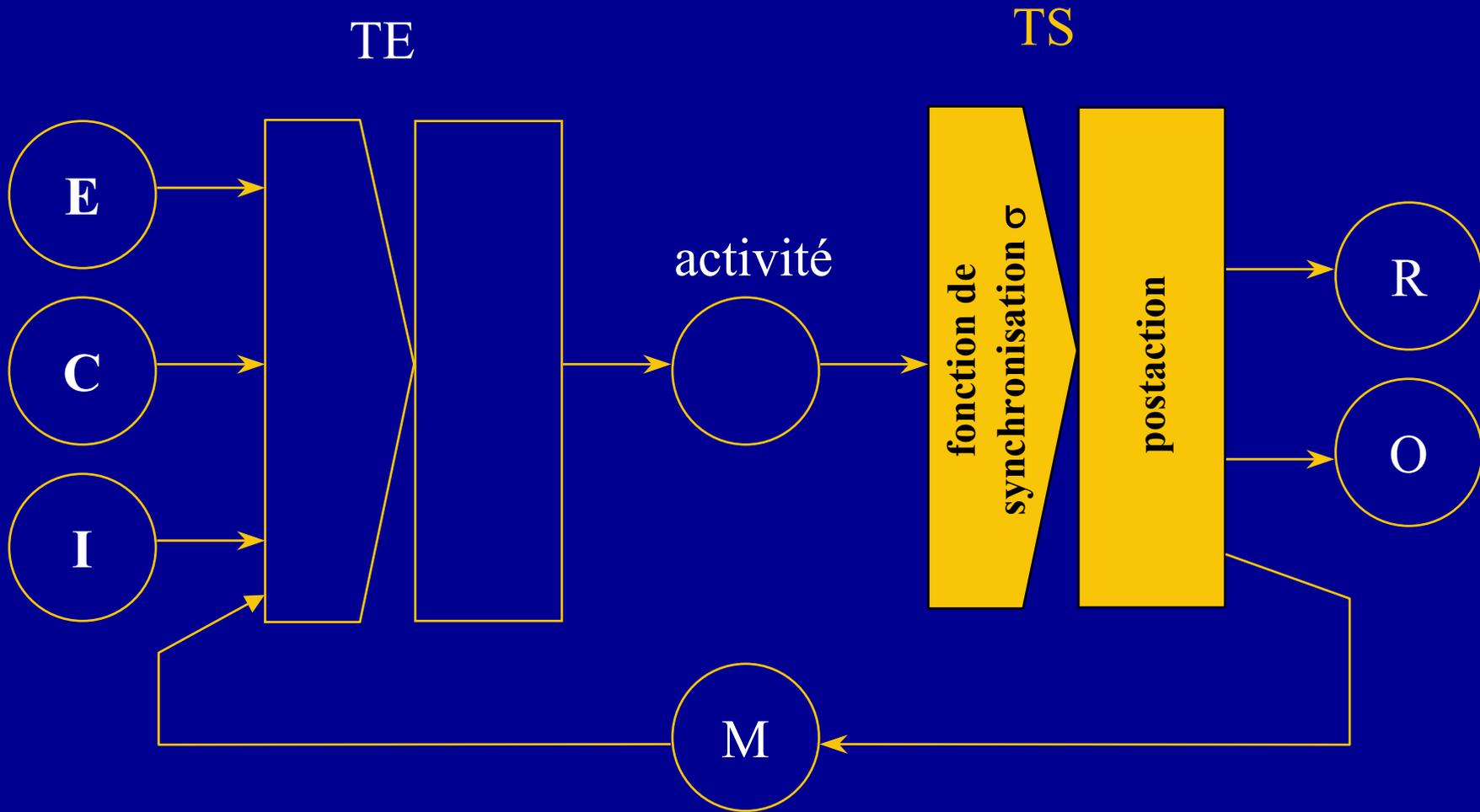


# *TCS : Place d'activité*



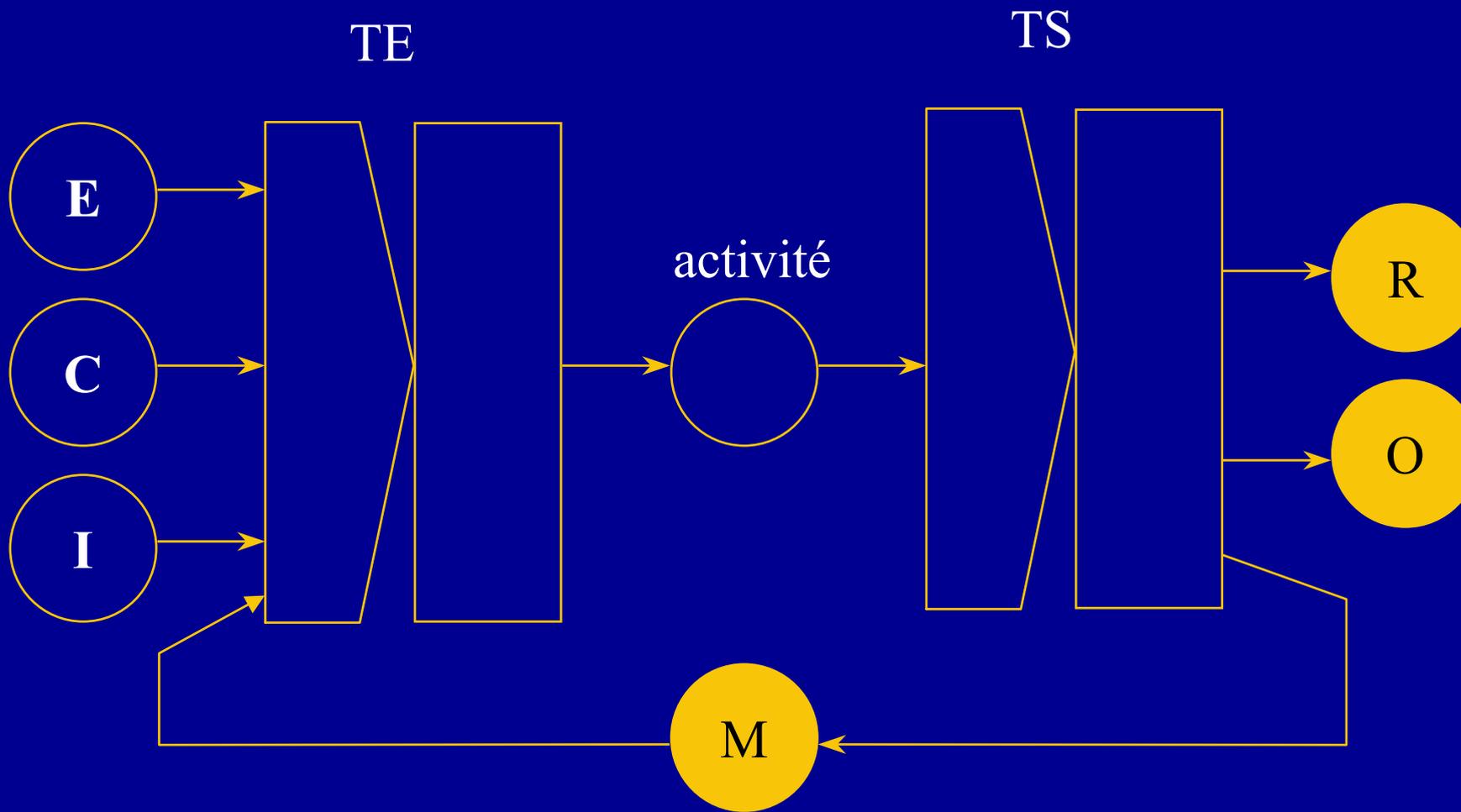


# TCS : Transition de sortie



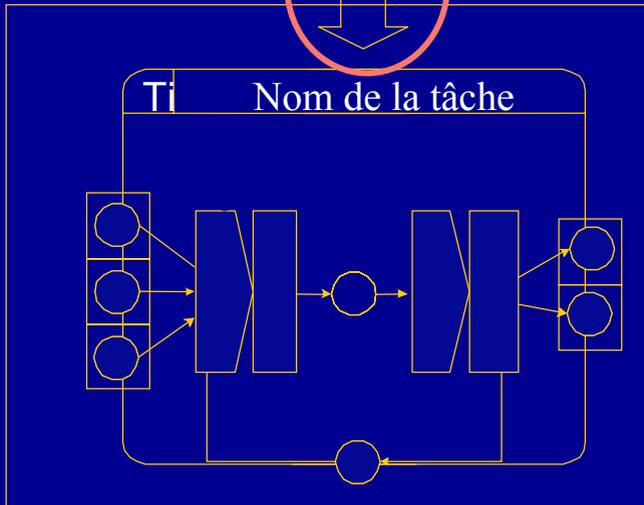
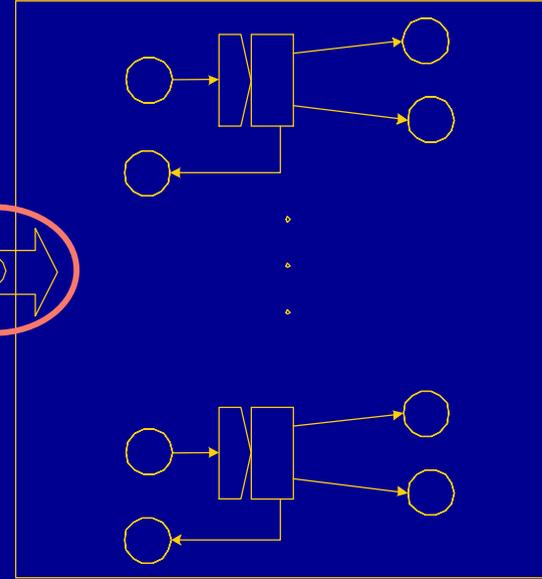
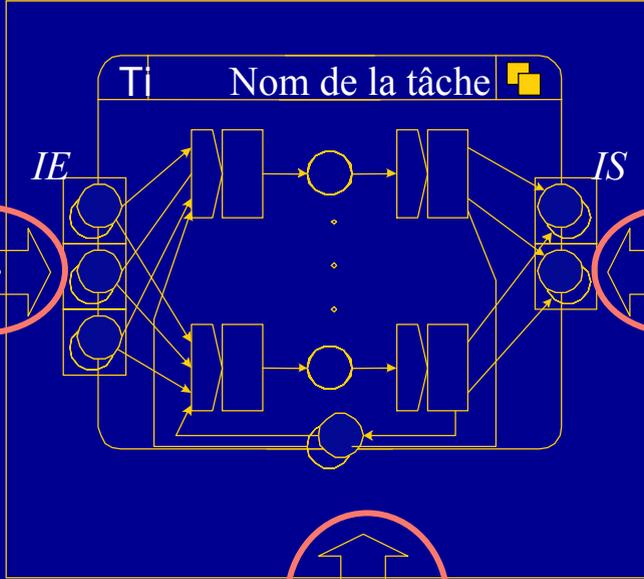
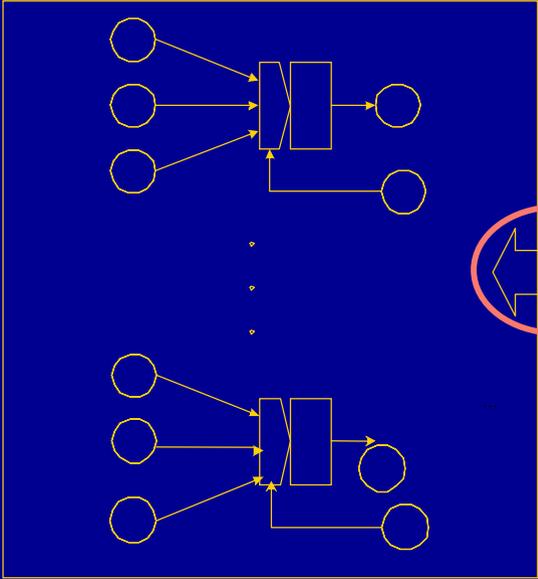


# TCS : Interface de sortie



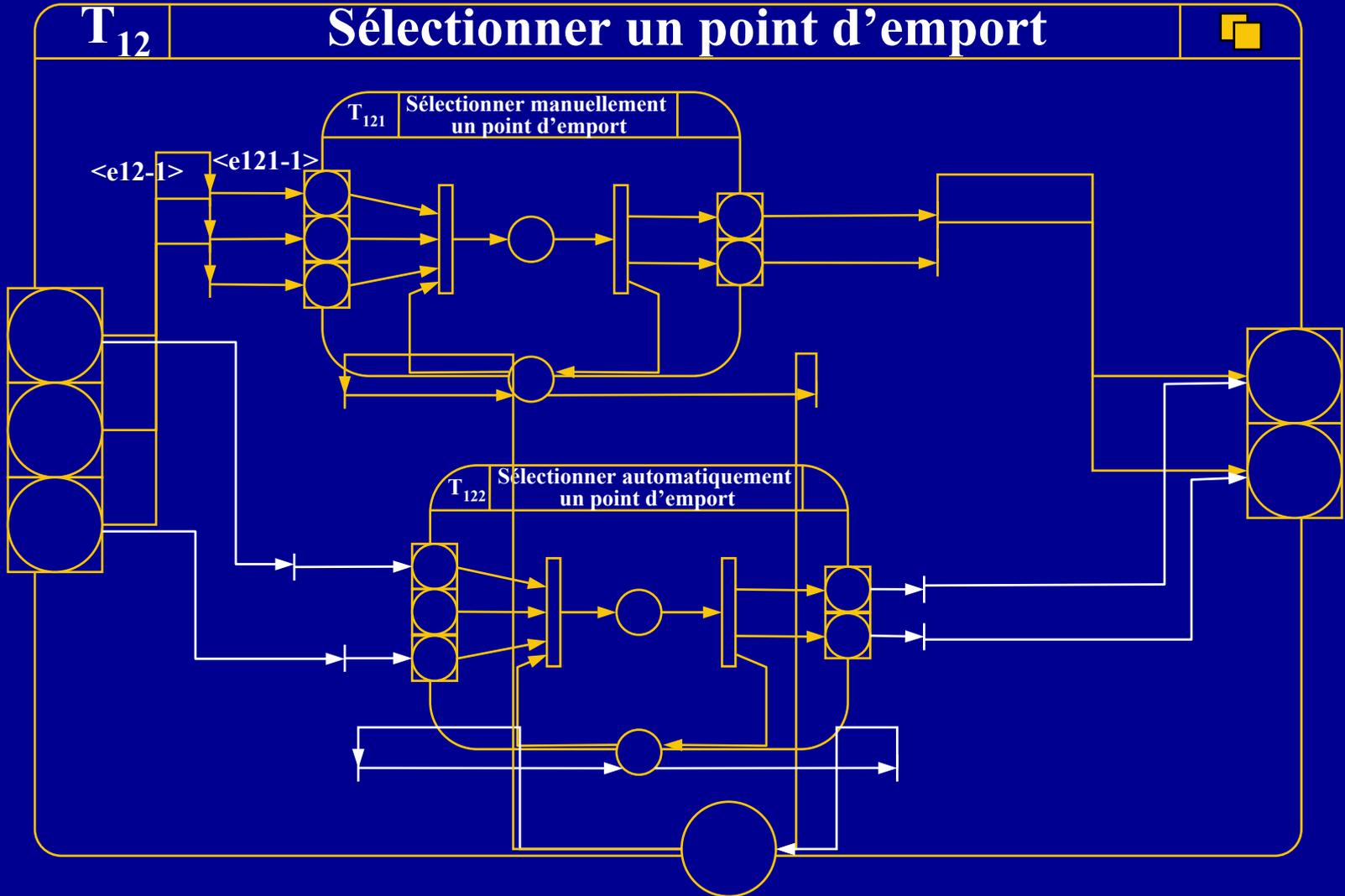


# De l'ObCS à la TCS





# Tâche T12 : Sélectionner PE



# Cycle de développement de TOOD

Analyse des besoins

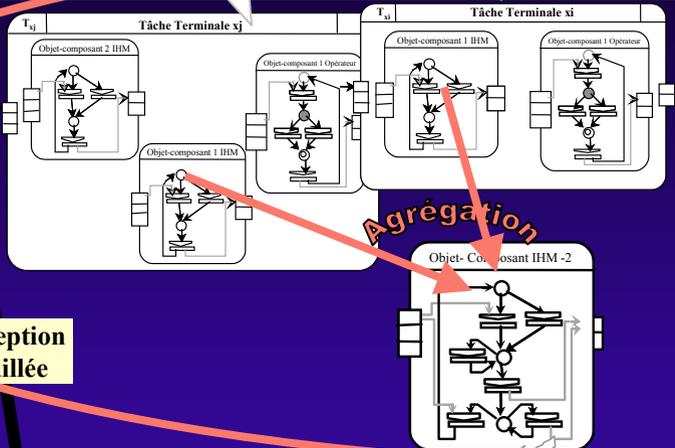
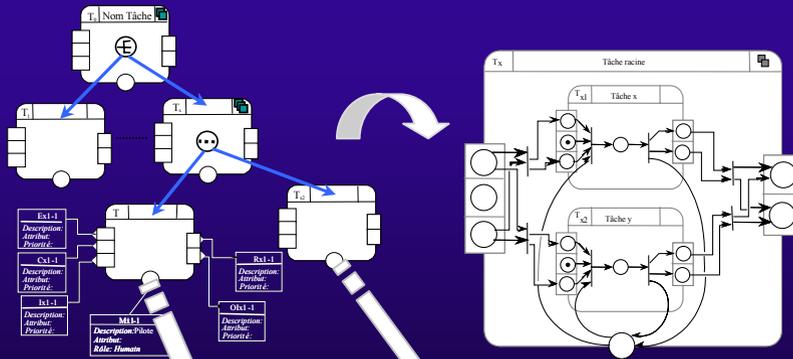
Spécification

Conception globale

Conception détaillée

Implémentation

Évaluation



Modèle des Objets du Domaine (MOD)

Modèle de la tâche

Modèle Statique de la Tâche (MST)

Modèle Dynamique de la Tâche (MDT)

Modèle opérationnel

Modèle Local de l'Interface (MLI)

Modèle de l'Utilisateur (MU)

Modèle Abstrait de l'Interface (MAI)

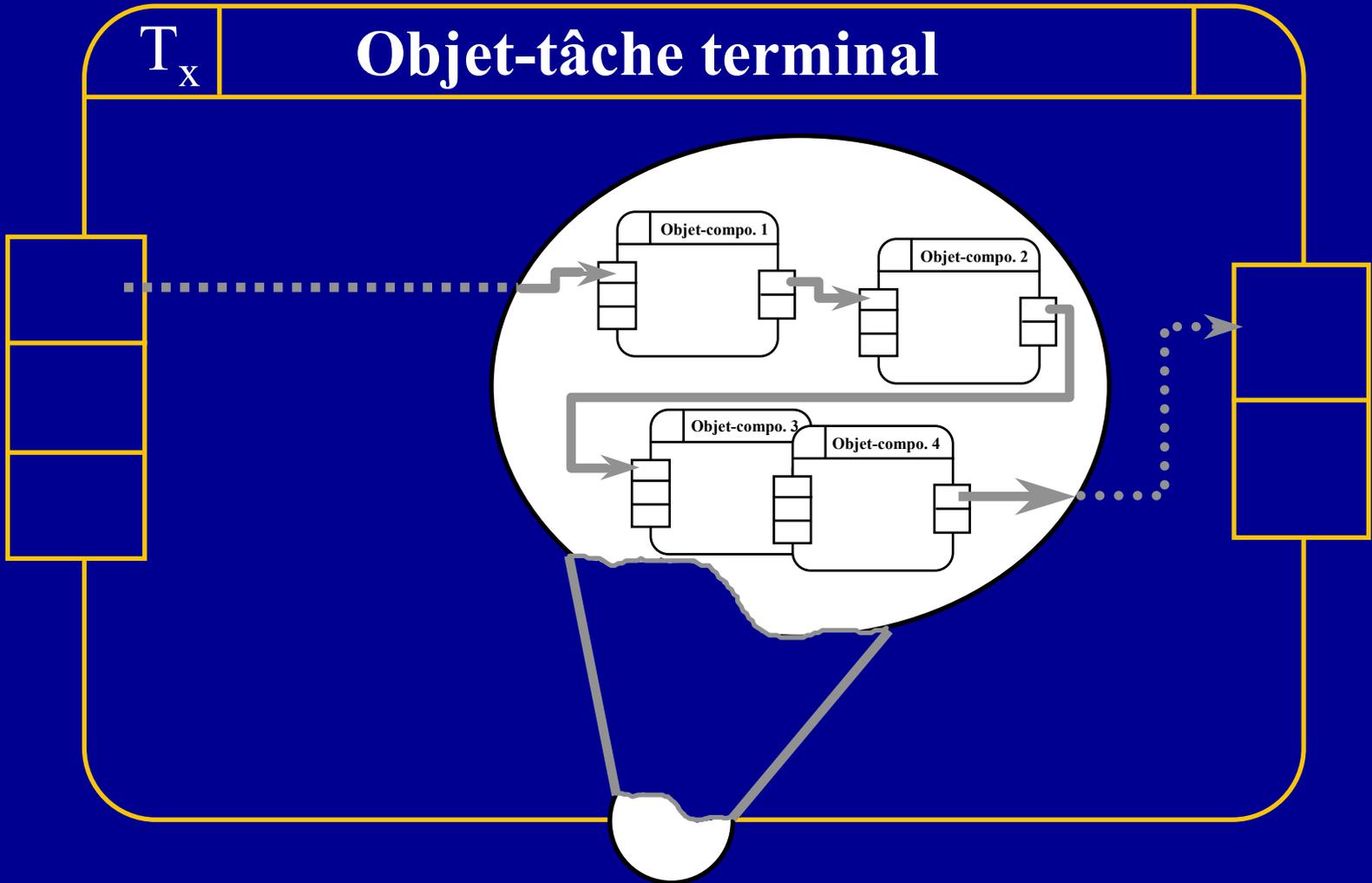
Modèle d'Implémentation de l'Interface (MII)

IHM

Évaluation

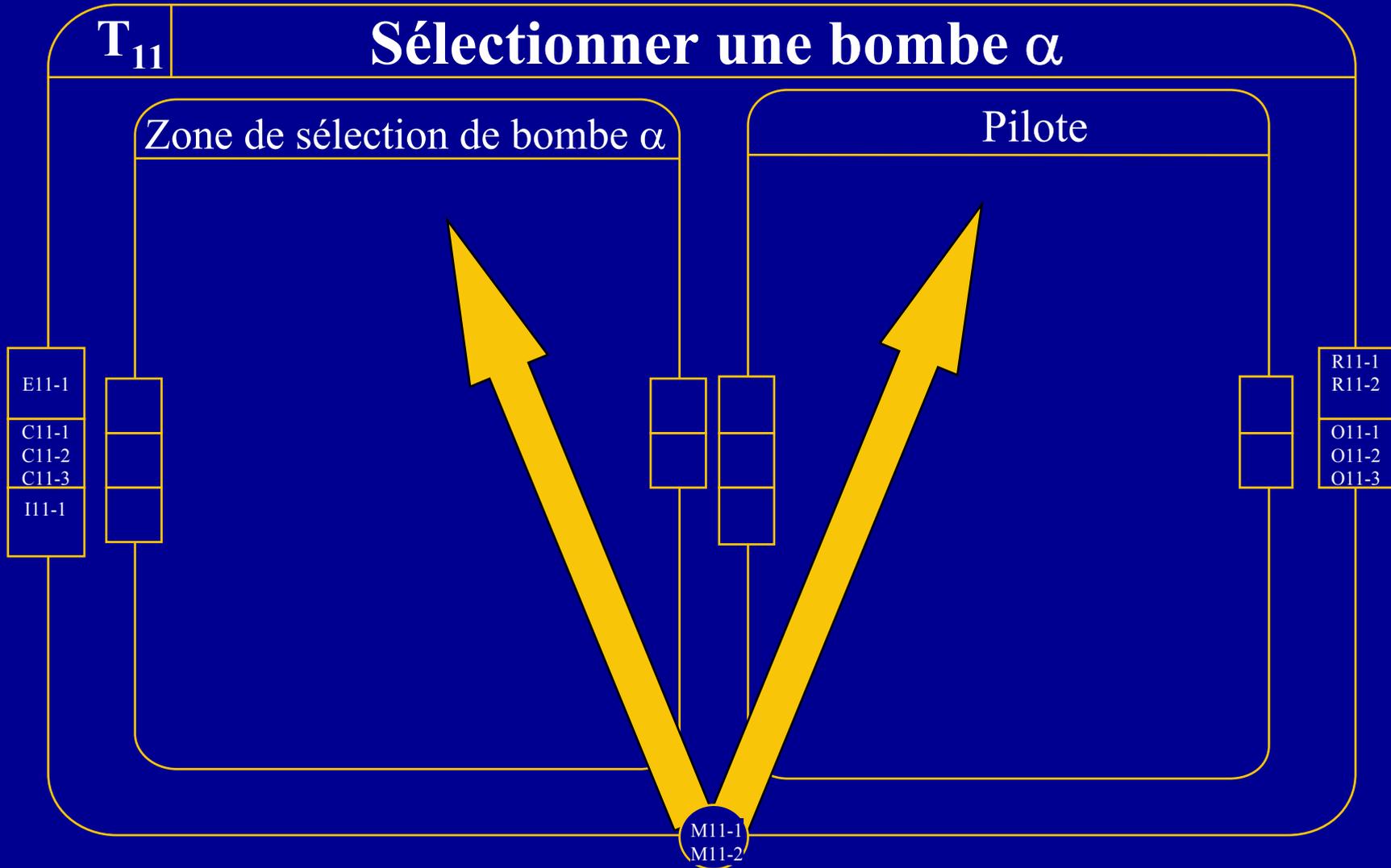


# Modèle opérationnel





# Modèle Opérationnel de $T_{11}$

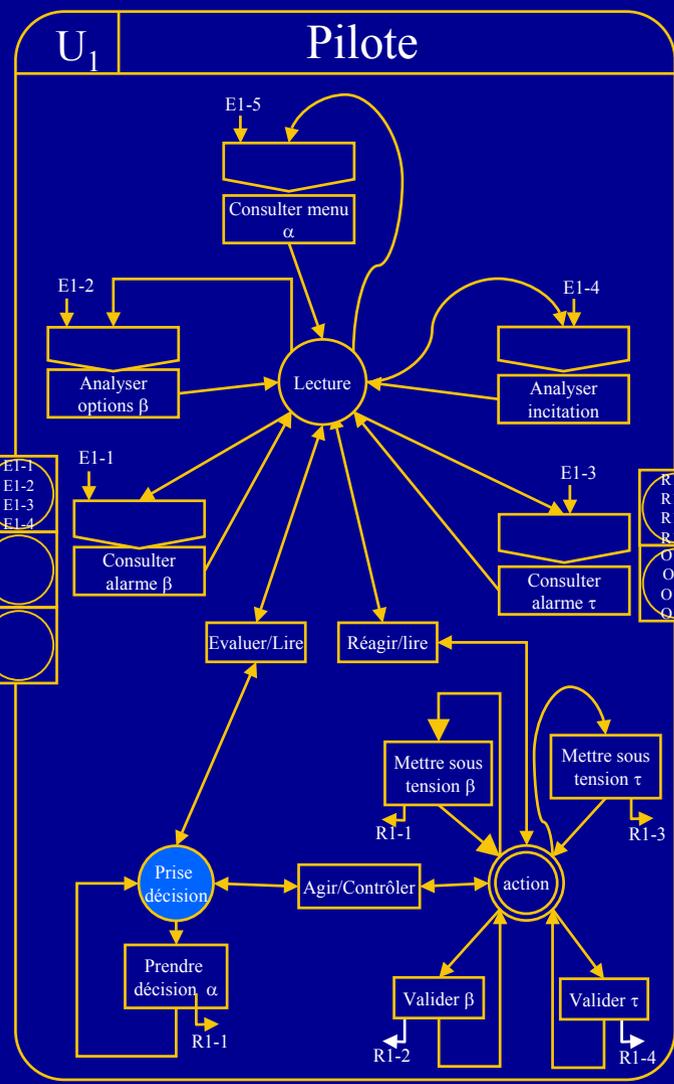




# Modèle Opérationnel de $T_{11}$

$T_{11}$

Sélectionner une bombe  $\alpha$



- E11-1
- C11-1
- C11-2
- C11-3
- C11-4
- I11-1
- I11-2

- E1-1
- E1-2
- E1-3
- E1-4

- R1-1
- R1-2
- R1-3
- R1-4
- O1-1
- O1-2
- O1-3
- O1-4

- R11-1
- R11-2
- O11-1
- O11-2



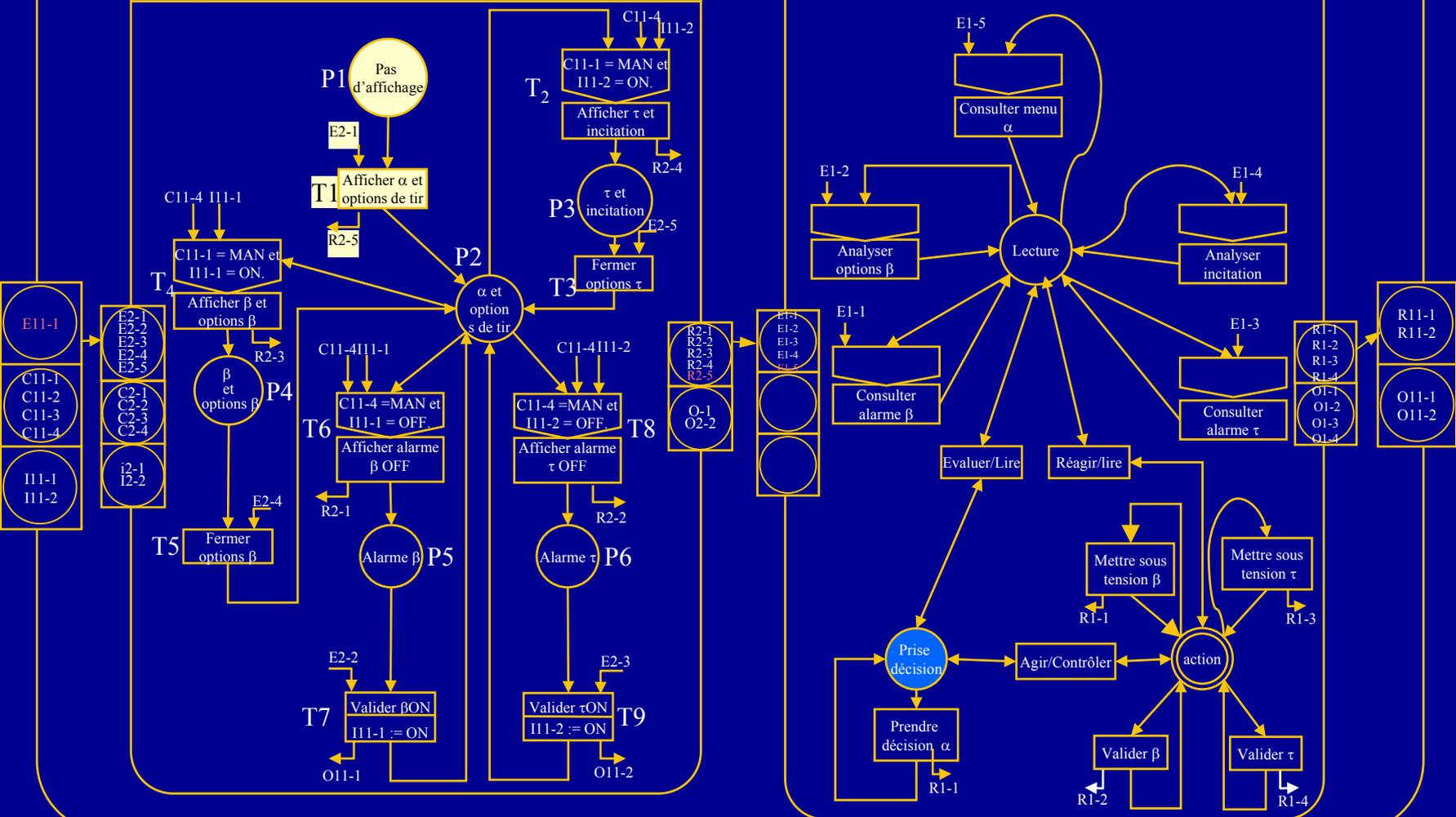
# Modèle Opérationnel de $T_{11}$

$T_{11}$

## Sélectionner une bombe $\alpha$

$OI_2$  Zone de sélection de bombe  $\alpha$

$U_1$  Pilote





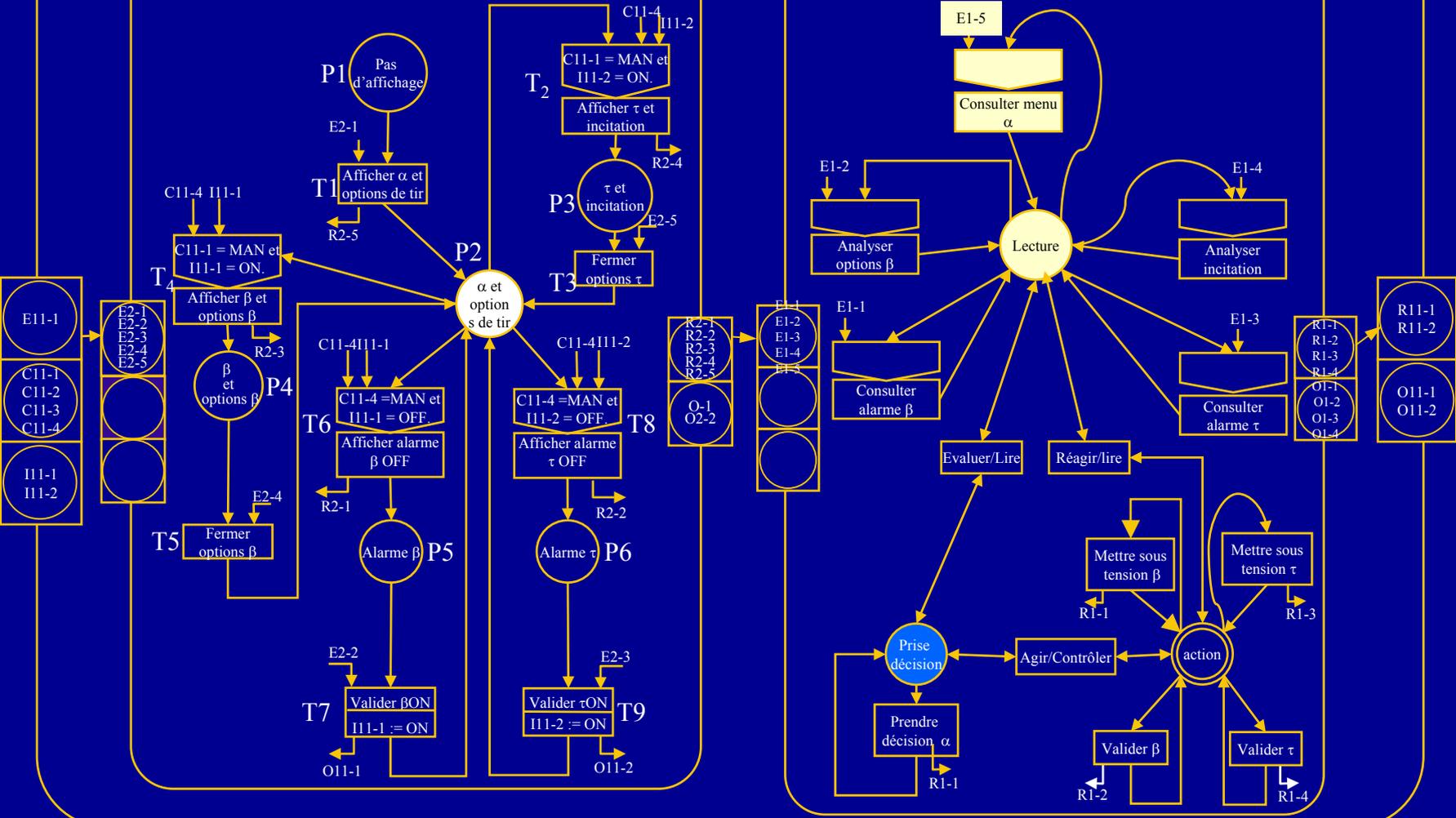
# Modèle Opérationnel de $T_{11}$

$T_{11}$

## Sélectionner une bombe $\alpha$

$OI_2$  Zone de sélection de bombe  $\alpha$

$U_1$  Pilote





# Modèle Opérationnel de $T_{11}$

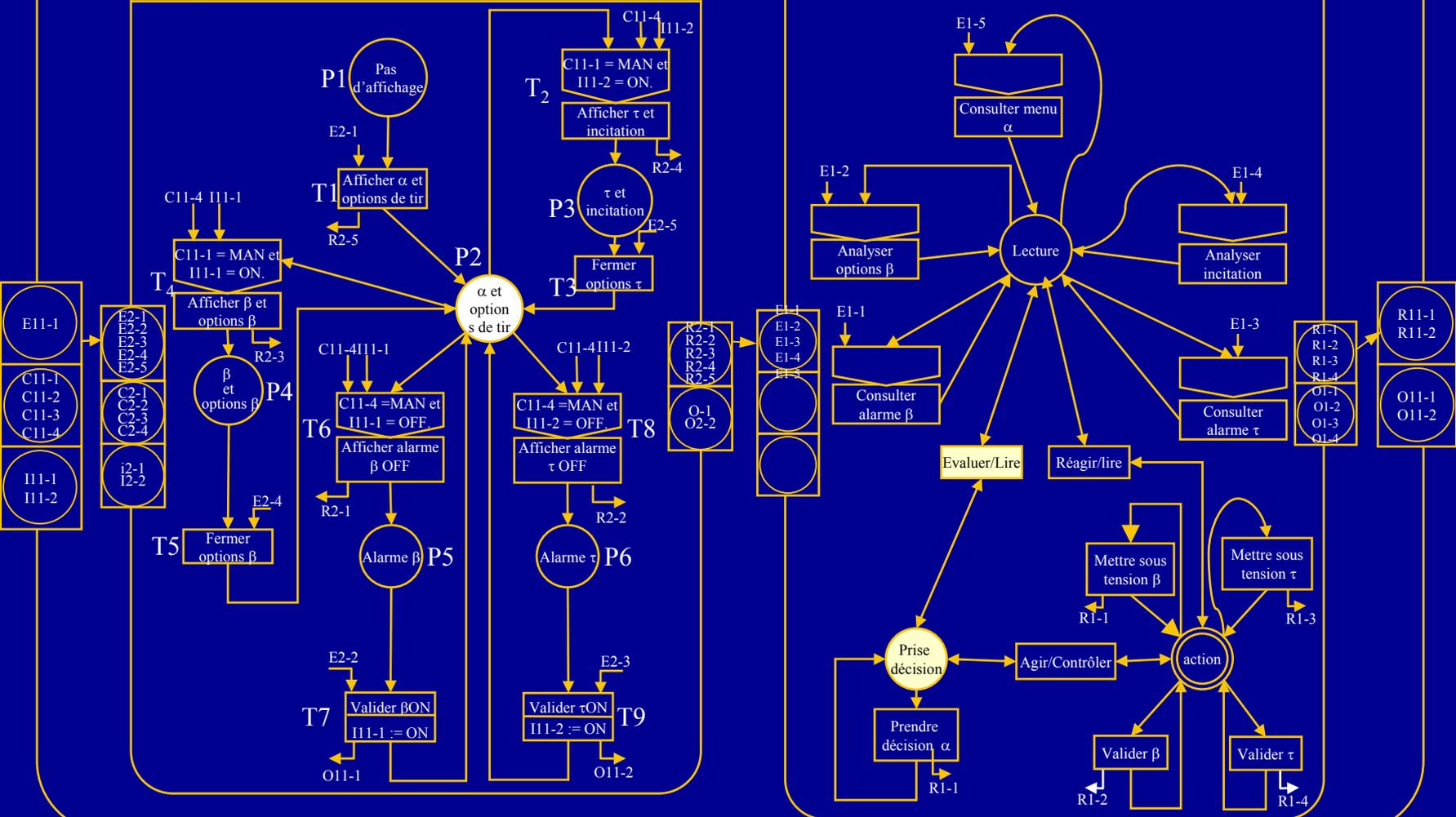
$T_{11}$

## Sélectionner une bombe $\alpha$

$OI_2$  Zone de sélection de bombe  $\alpha$

$U_1$

Pilote





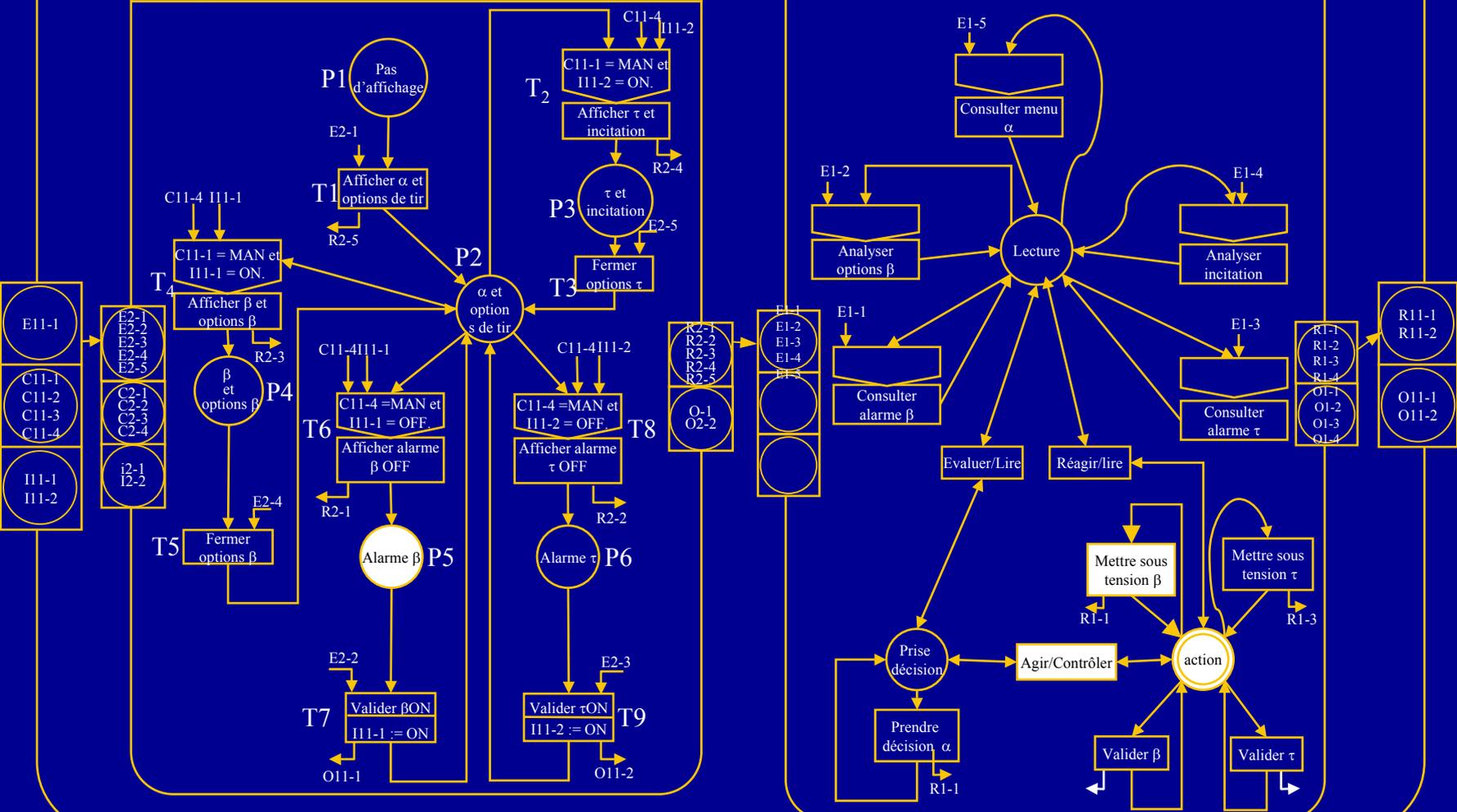
# Modèle Opérationnel de $T_{11}$

$T_{11}$

## Sélectionner une bombe $\alpha$

$OI_2$  Zone de sélection de bombe  $\alpha$

$U_1$  Pilote





# Modèle Opérationnel de $T_{11}$

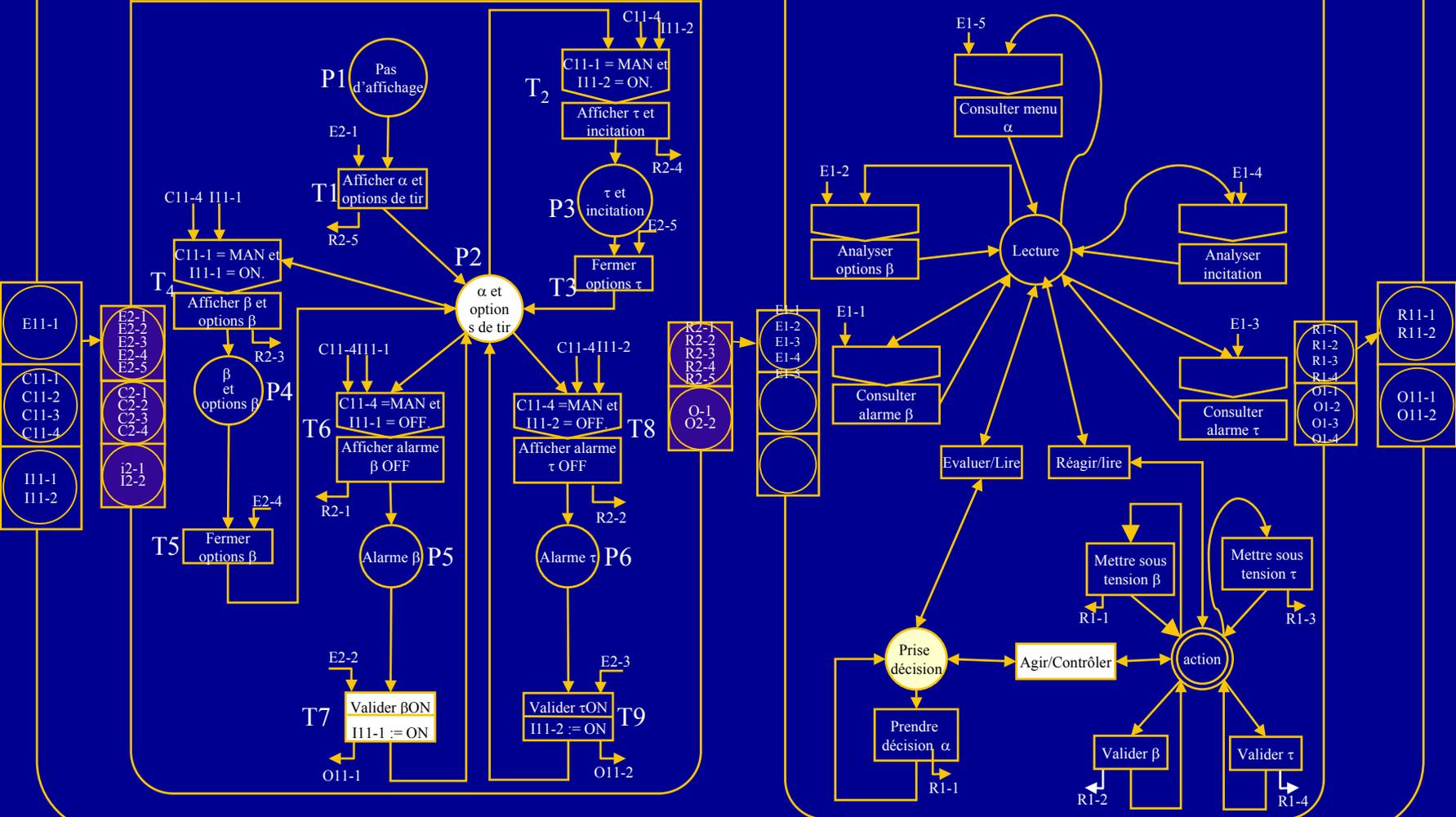
$T_{11}$

## Sélectionner une bombe $\alpha$

$OI_2$  Zone de sélection de bombe  $\alpha$

$U_1$

Pilote





# Modèle Opérationnel de $T_{11}$

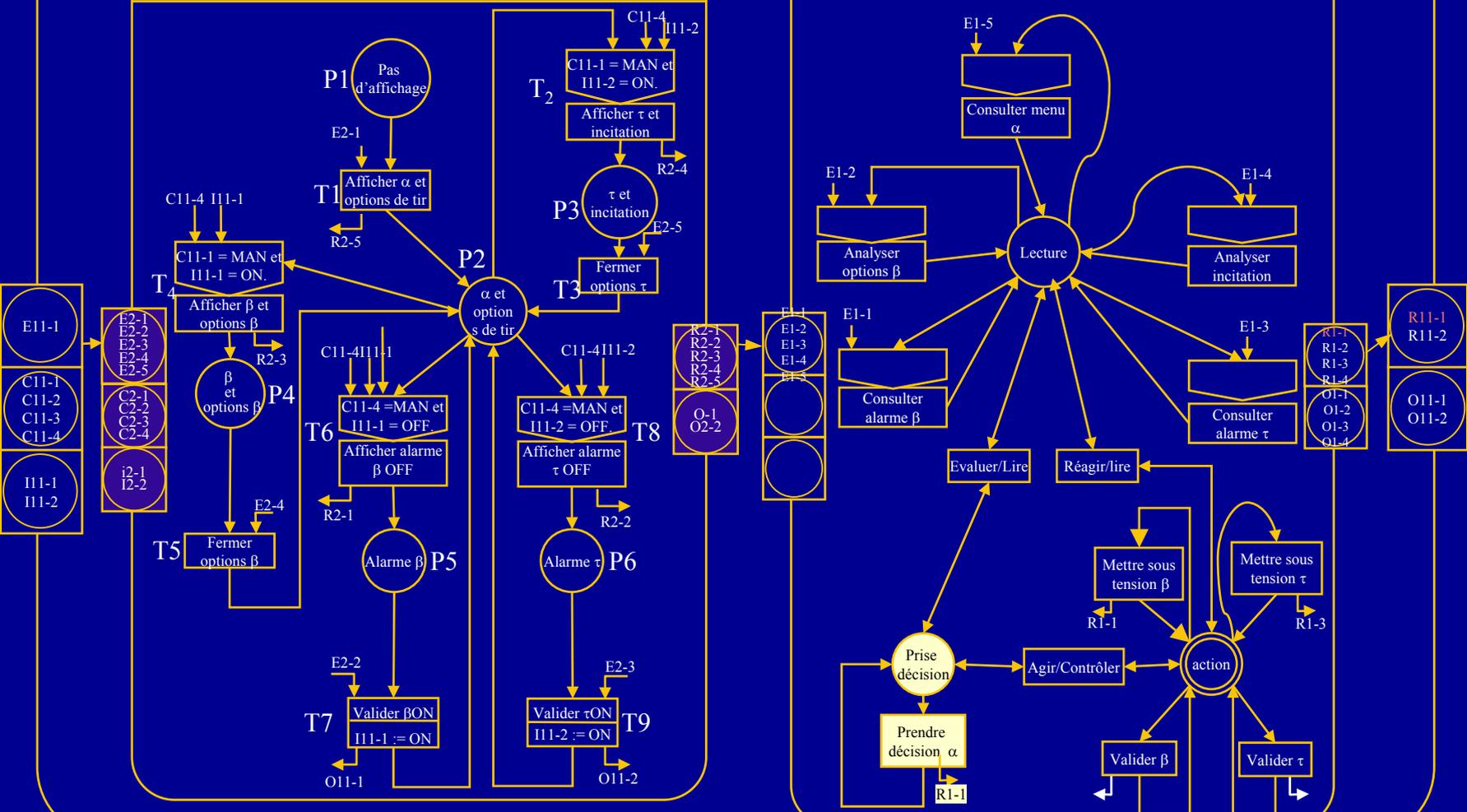
$T_{11}$

## Sélectionner une bombe $\alpha$

$OI_2$  Zone de sélection de bombe  $\alpha$

$U_1$

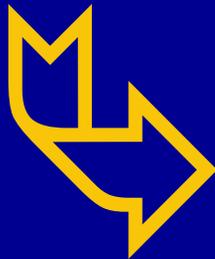
Pilote



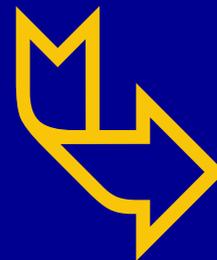


# *Modèle Abstrait de l'Interface*

**Modèle Local  
de l'Interface**  
Spécification  
des objets-interactifs (IHM)  
de chaque objet-tâche terminal



**Mécanisme  
d'Agrégation**  
Duplication des objets-  
interactifs



**Modèle Abstrait  
de l'Interface**  
Spécification de haut  
niveau de l'interface finale



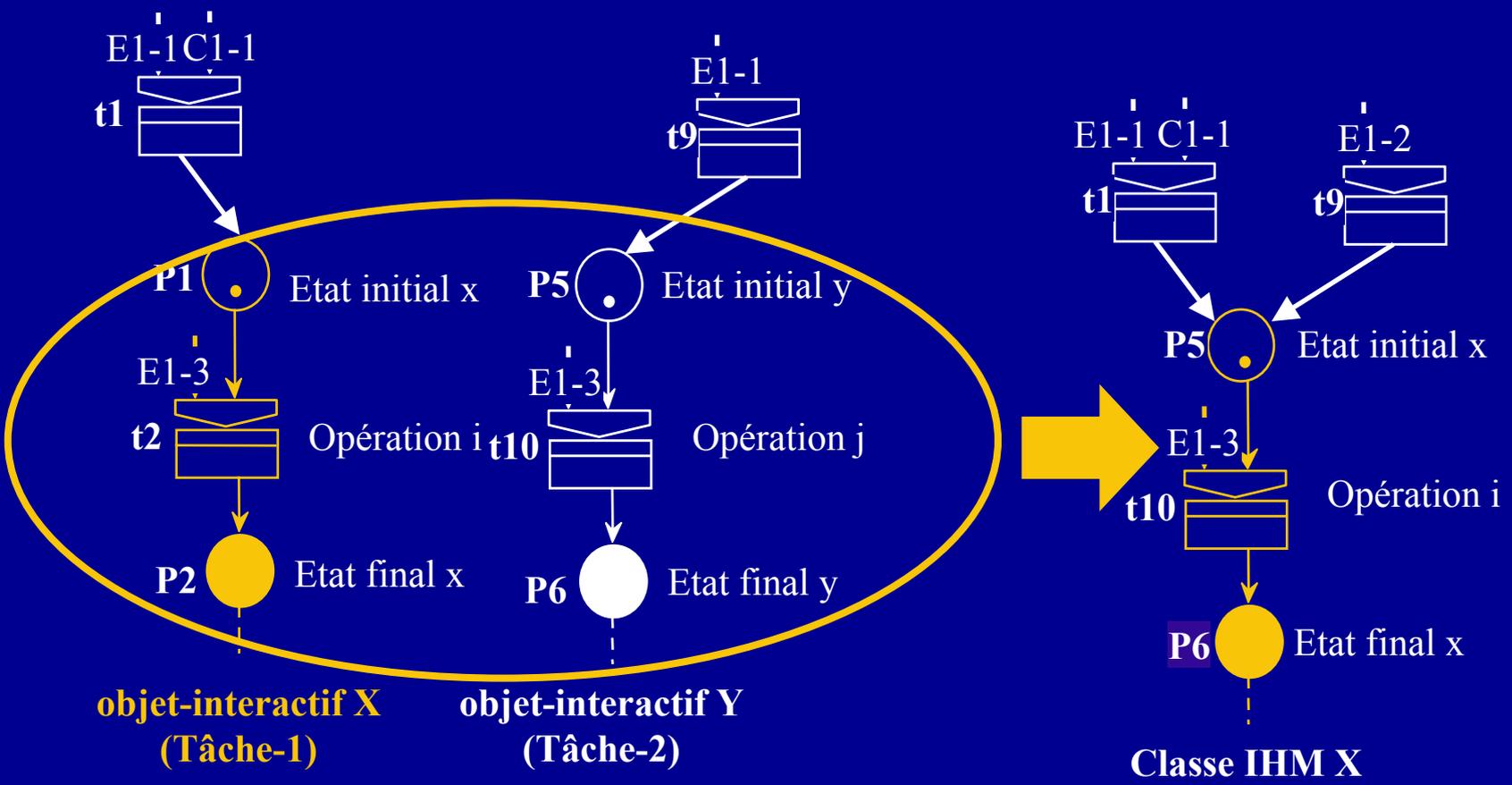
# Fusion de transitions

Construction de l'ObCS d'une classe IHM par **fusion de transitions**

↳ *Composition par fusion de transitions :*

fusion des transitions modélisant une même opération et une même procédure.

=> Si (état initial x = état initial y ET état final x = état final y ET Opération i = Opération j)

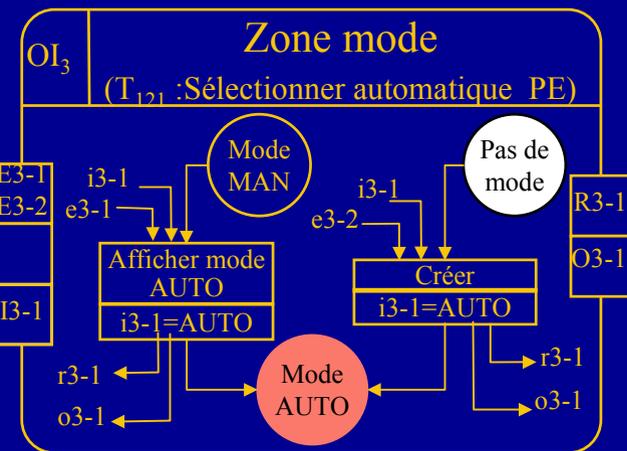
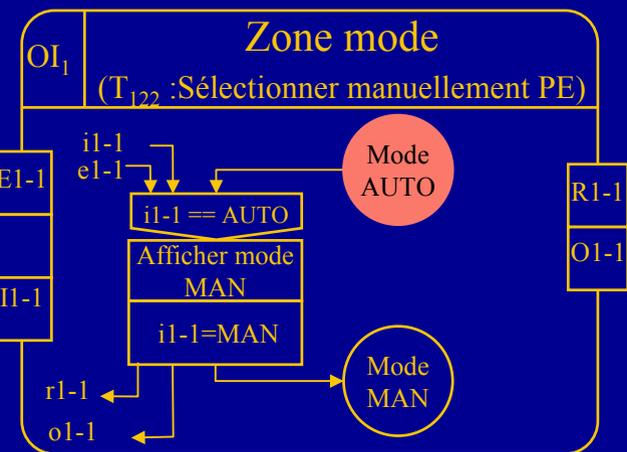




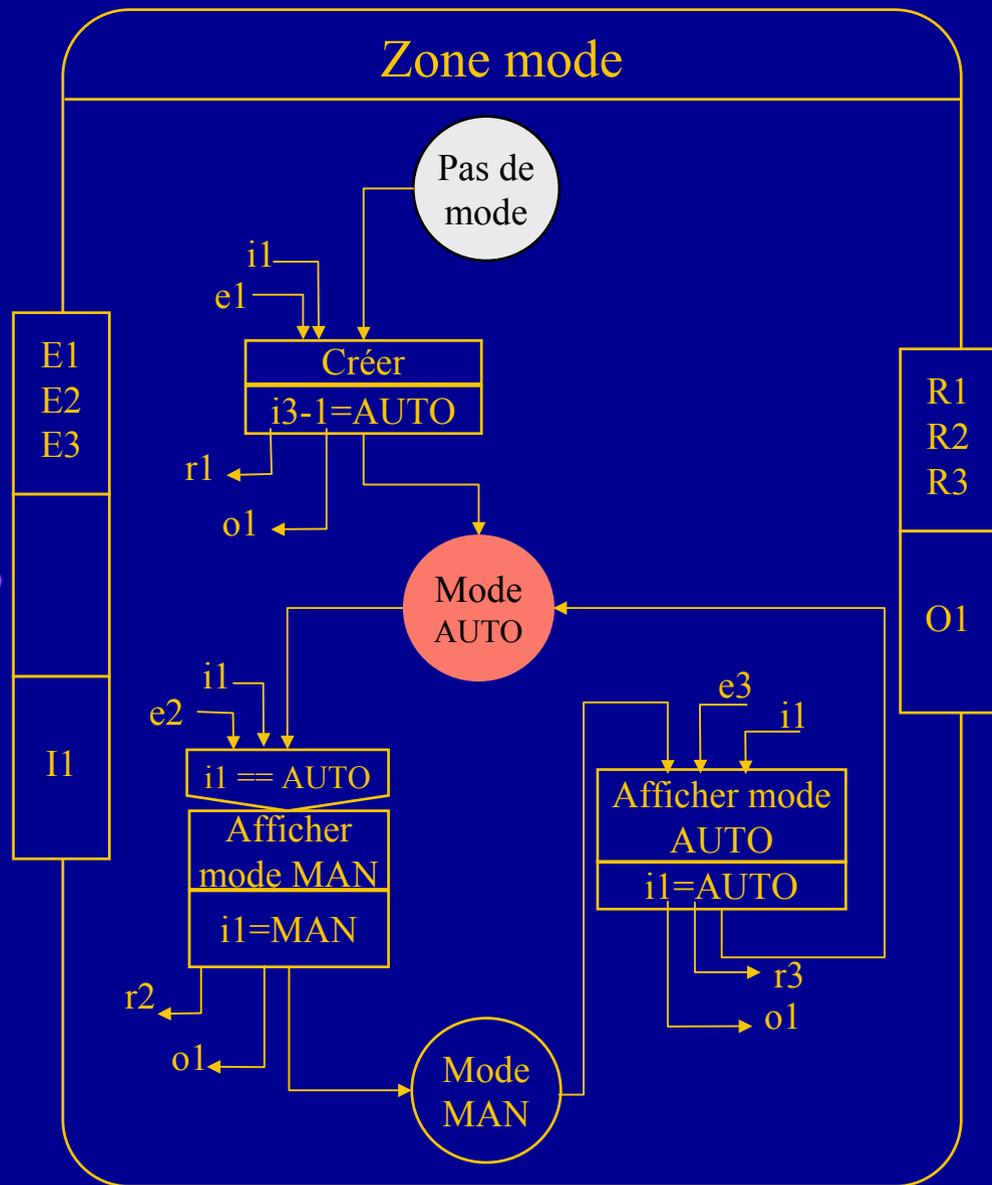
# Zone mode ( $T_{121}$ et $T_{122}$ )

Objet-tâche Terminal Objet Interactif	$T_{11}$ Sélectionner une bombe $\alpha$	$T_{121}$ Sélectionner automatiquement le point d'emport	$T_{122}$ Sélectionner manuellement le point d'emport	$T_{13}$ Sélectionner les fusées	$T_2$ Préparer les options de conduite de tir
Zone de sélection bombe $\alpha$	✓				
Zone mode		✓	✓		
Zone sélection point d'emport			✓		
Zone de sélection fusée				✓	
Zone de sélection de conduite					✓

# Agrégation de la zone mode



Agrégation



# Cycle de développement de **TOOD**



Analyse des besoins

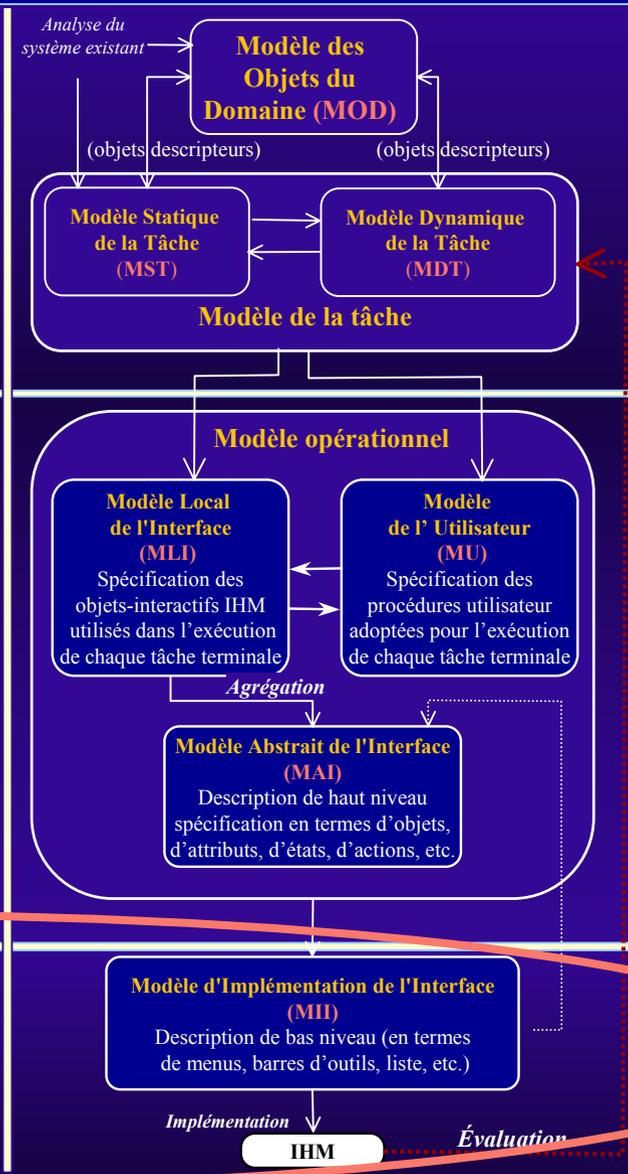
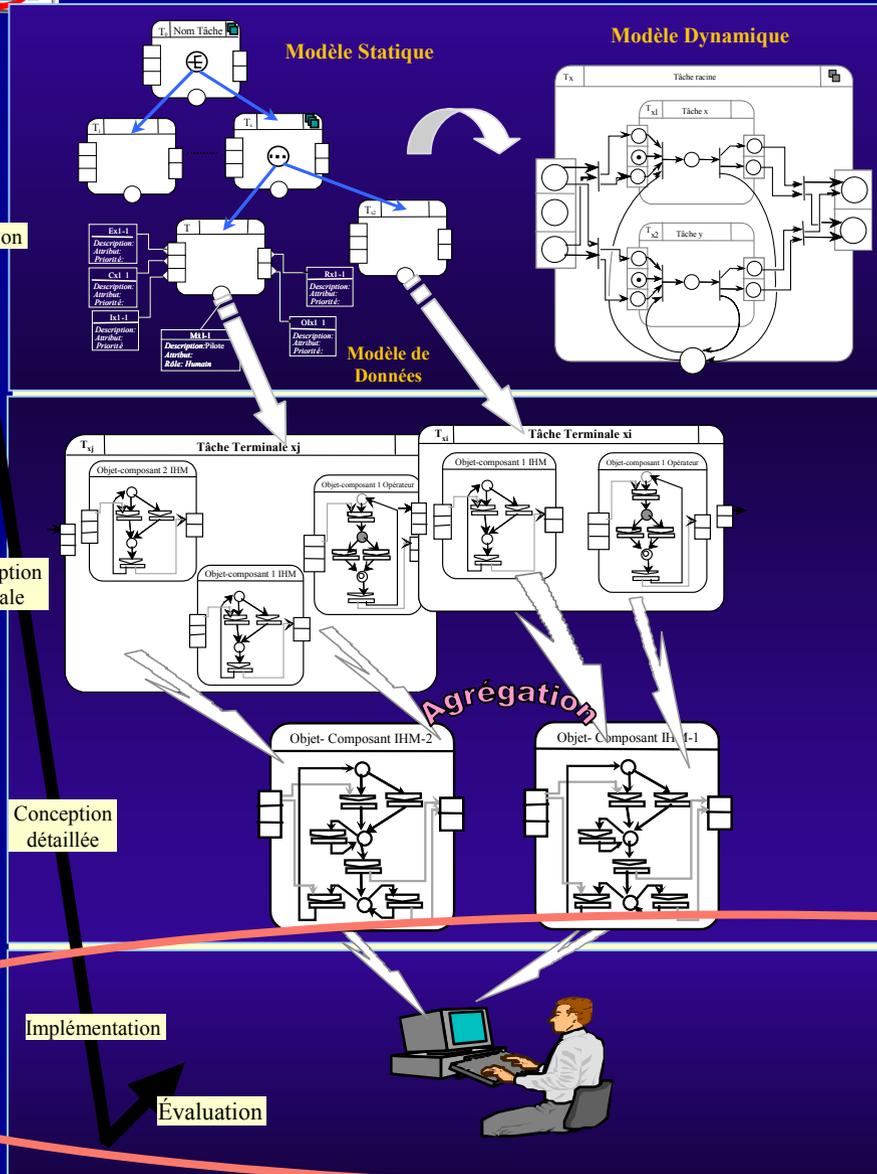
Spécification

Conception globale

Conception détaillée

Implémentation

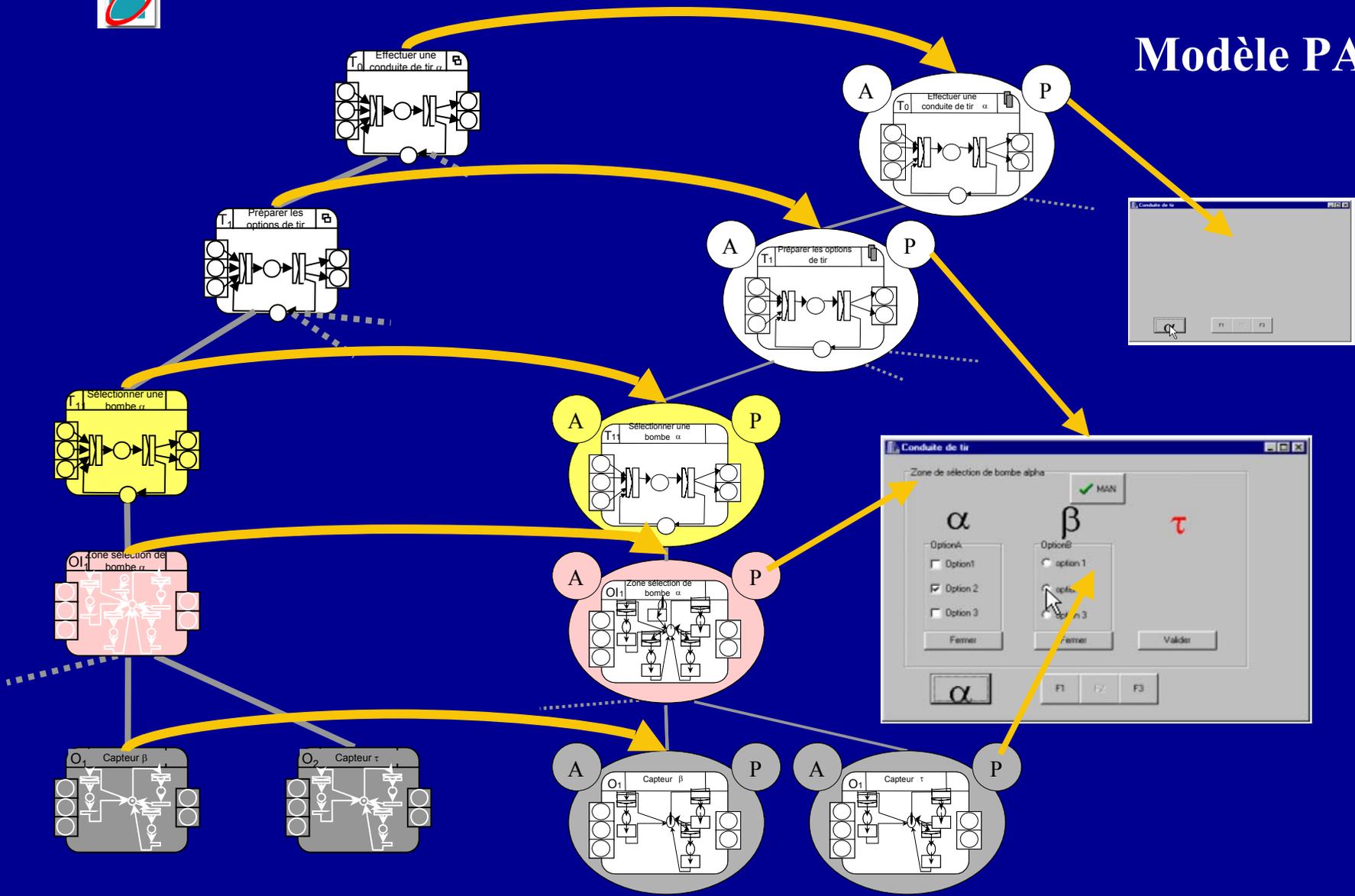
Évaluation



# Construction de la hiérarchie d'agents



Modèle PAC



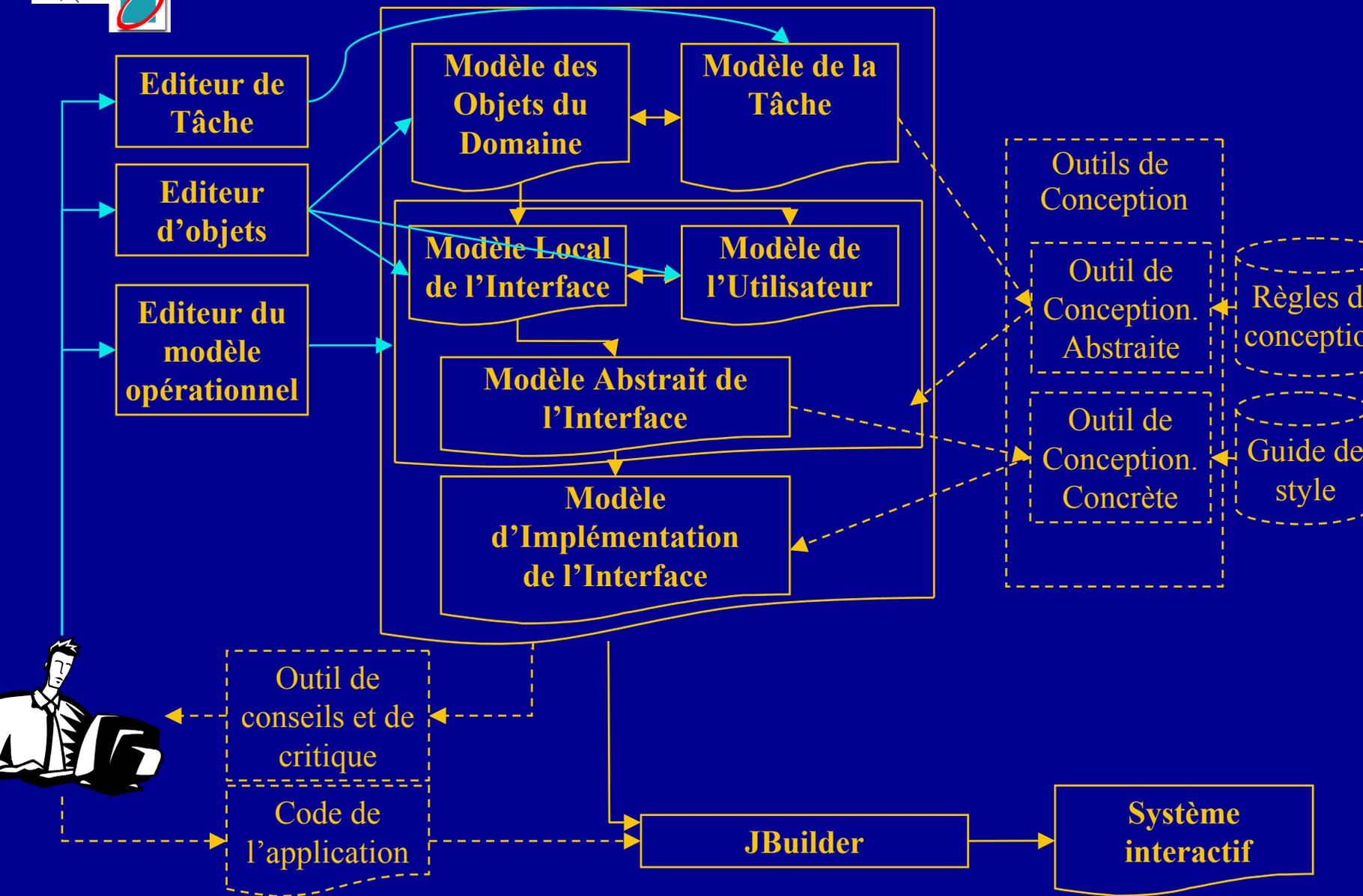


# Plan

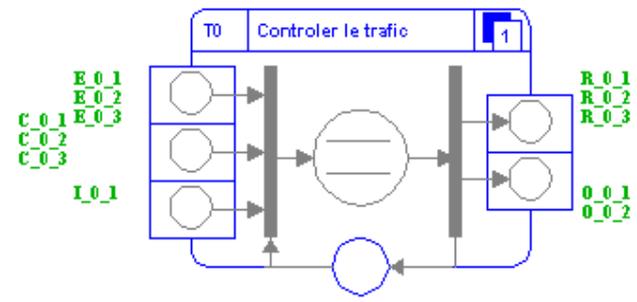
---

1. **Cycles de développement et outils de base en conception des IHM**
  2. **TOOD : Modèle générique et formalismes de base**
  3. **TOOD : Démarche méthodologique**
  4. **TOOD : Environnement de développement**
- C&P Conclusion et perspectives de recherche**

# TOOD : Environnement MB-IDE



- T0 : Contrôler le trafic
  - T1 : Planifier le trafic
    - T11 : Configurer l'entrée d'un vol
      - T111 : Connaître le NV
      - T112 : Intégrer le NV
        - T1121 : Analyser les CE
        - T1122 : Prendre décision CE
    - T12 : Gérer le trafic



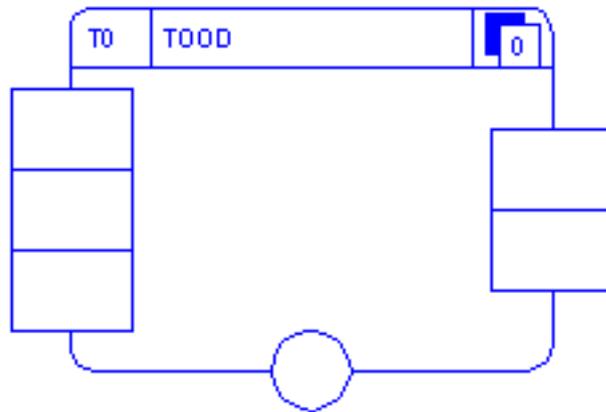


# TOOD-IDE

Editeur TOOD

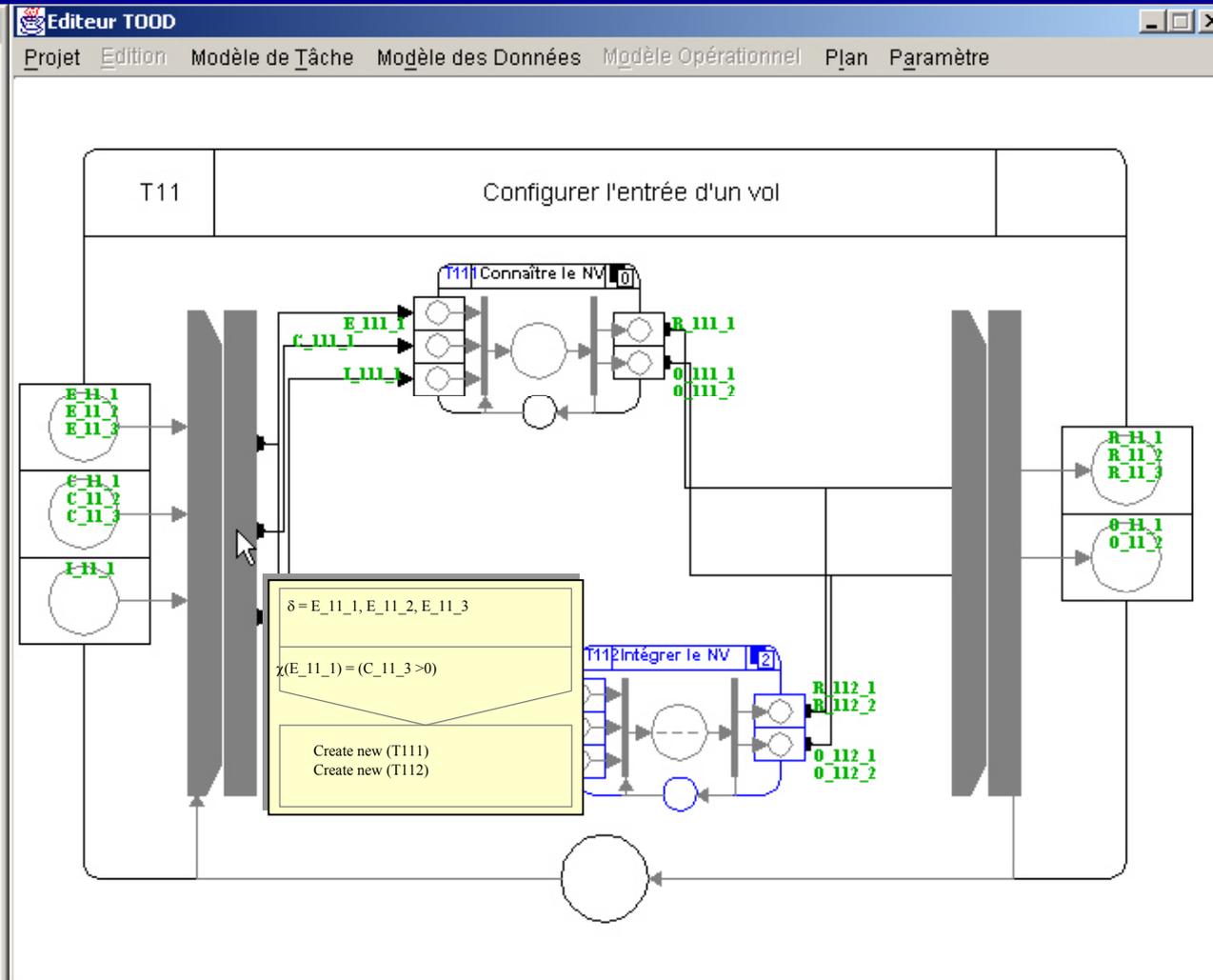
Projet Edition **Modèle de Tâche** Modèle des Données Modèle Opérationnel Plan Paramètre

- Decomposer
- Spécifier
- Définir TCS
- Générer
- Simuler
- Imprimer Spécification
- Imprimer Code





# Editeur du modèle de la tâche





# Spécification de la tâche (1)

**Configurer l'entrée d'un vol**

Tâche | Enclenchements | Contrôles | Entrées | Réactions | Sorties | Ressources

Nom: Configurer l'entrée d'un vol

Description: Préparer le trafic aérien à l'entrée du secteur

- T11 : Configurer l'entrée d'un vol
  - T111 : Connaître le NV
  - T112 : Intégrer le NV
    - T1121 : Analyser les CE
    - T1122 : Prendre décision CE

Fermer | Annuler | Générer | Ajouter | Supprimer | Remplacer | Importer

**Configurer l'entrée d'un vol**

Tâche | Enclenchements | Contrôles | Entrées | Réactions | Sorties | Ressources

Identificateur	Type	Description
E_11_4		

E\_11\_1:=E\_1\_1:Arrivée d'un nouveau vol  
E\_11\_2:=E\_1\_2:Proposition du niveau d'entrée (EFL) du secteur précédent  
E\_11\_3:=E\_1\_3:Warning sur EFL

Fermer | Annuler | Générer | Ajouter | Supprimer | Remplacer | Importer

# TCS : constructeur

TCS

Pré Action    Synchronisation    Complétude Sortie    Post Action

Constructeur    Priorité    Complétude Entrée    Cohérence Entrée

TCS Terminale

TCS Séquentielle

TCS Parallèle

TCS Choix

Terminée



# TCS : onglet Complétude Entrée

The screenshot shows the TCS software interface with the 'Complétude Entrée' tab selected. The window title is 'TCS'. The interface is divided into several sections:

- Top Navigation:** A row of tabs: 'Pré Action', 'Synchronisation', 'Complétude Sortie', and 'Post Action'. Below these are sub-tabs: 'Constructeur', 'Priorité', 'Complétude Entrée', and 'Cohérence Entrée'. The 'Priorité' sub-tab is currently active, and a mouse cursor is pointing at it.
- Central Area:** A large white area with a grey border. On the right side, there is a list of items: 'C\_11\_2:Temps de temporisation' and 'I\_11\_1:Niveau de vol en entrée'. A scroll bar is visible below the list. In the center, there are two buttons: a top button with a right-pointing arrow and a bottom button with a left-pointing arrow.
- Bottom Area:** A row of buttons: 'Précedent', 'Suivant', and 'Terminée'. To the left of the 'Précedent' button, the text 'E\_11\_1:=E\_1\_1...' is visible.



# Editeur des objets

Choisir un agent dans la liste ou en éditer un nouveau:

strip

Editer    Générer

Abstraction   **Présentation**   Contrôle

Attributs:

Nom	Type	Defaut	get	set	Visible
AFL	Integer		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Indicatif	String		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Type	Integer		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
CFL	Integer		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Sfl	Integer		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
RFL	Integer		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Editer Attribut

Méthodes:

Nom de la Methode	Type retourné
ZoomRoute	Boolean
ZoomIndicatif	Integer

Editer Méthode    Editer le corps de la méthode

Choisir un agent dans la liste ou en éditer un nouveau:

strip

Editer    Générer

Abstraction   **Présentation**   Contrôle

- Ajouter une place
- Ajouter une transition
- Ajouter une trans. Enclenchement
- Ajouter une trans. Réaction

```
graph TD; P1(( )) --> C[Créer]; C --> P2(( )); P2 --> ZR[Zoom Route]; P2 --> AW[Afficher Warning]; ZR --> P3(( )); AW --> P3; P3 --> P2; E11[E_1_1] --> ZR; ZR --> R11[R_1_1];
```



# Code Java généré

pcGRASP - [T11Configurelntreedunvol.java]

File Edit View Templates Project Compiler Run Windows Help

Generate CSD Remove CSD Font Size: 8

```
import java.util.*;/**
**
**
public class T11Configurelntreedunvol extends MetaTache {
/**
*Arrivée d'un nouveau vol
**
- public DonnéeT00D E_11_1=DonnéeT00D.forType("Arrivee_NV");
/**
*Proposition du niveau d'entrée (EFL) du secteur précédent
**
- public DonnéeT00D E_11_2=DonnéeT00D.forType("Proposition_EFL_SP");
/**
*Warning sur EFL
**
- public DonnéeT00D E_11_3=DonnéeT00D.forType("Warning_EFL");
/**
*Renseignements généraux sur le vol
**
- public DonnéeT00D C_11_1=DonnéeT00D.forType("Renseignements_generaux");
/**
*Conditions météo
**
- public DonnéeT00D C_11_2=DonnéeT00D.forType("Conditions_meteo");
/**
*Temps de temporisation
**
- public DonnéeT00D C_11_3=DonnéeT00D.forType("Temps_Temporisation");
/**
*Niveau de vol en entrée
**
- public DonnéeT00D I_11_1=DonnéeT00D.forType("EFL");
/**
*vol planifié en entrée
**
}
```

pcGRASP Starting CSD Generation ...  
pcGRASP CSD Generation Complete

For Help, press F1

Ln 4, Col 60

# Simulateur du modèle de la tâche



**Simulateur**

T0 Contrôler le trafic | T11 Configurer l'entrée d'un vol | T1 Planifier le trafic | T111 Connaître le NV

Transition d'entrée

<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
E1	C1
etat <input type="text" value="true"/>	Vent <input type="text" value="100"/> Nebula... <input type="text" value="10"/>
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
E2	C2
Etat <input type="text" value="true"/>	second... <input type="text" value="30"/> minutes <input type="text" value="1"/>
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

**Arbre**

- T0 : Contrôler le trafic
  - T1 : Planifier le trafic
    - T11 : Configurer l'entrée d'un vol
      - T111 : Connaître le NV
      - T112 : Intégrer le NV
        - T1121 : Analyser les CE
        - T1122 : Prendre décision CE

**Editeur TOOD**

Projet | Edition | Modèle de Tâche | Modèle des Données | Modèle Opérationnel | Plan | Paramètre

T11 Configurer l'entrée d'un vol



# Conclusion

## ANALYSE

Modèles	Notation	Supports
MT, MOD	RdPO	Editeur du modèle de tâche, éditeur d'objets

## CONCEPTION

Modèles	Notation	Guide de conception	Supports
MLI, MU, MAI	RdPO	<i>En perspective</i>	Editeur d'objet, Editeur MO en perspective

## IMPLEMENTATION

Types de génération	Résultats de génération	Guide de style	Outil de conseils et de critique
Automatique pour le dialogue, manuelle pour la présentation	Dialogue, Présentation (présentation en perspective)	<i>En perspective</i>	<i>En perspective</i>

## Cas de validation

**Application bureautique, Contrôle aérien, Conduite de tir**



# *Perspectives liées à la méthode*

---

- ◆ **Vers une génération semi-automatique de la présentation**
- ◆ **Vers une vérification automatique des modèles**
  - **Structurelle**
  - **Comportementale par une simulation de scénario**
- ◆ **Vers une spécification multi-langage et multi-plateforme** [Thévenin, 2001], [Eisenstein et al., 2001]
- ◆ **Vers une assistance «intelligente»**

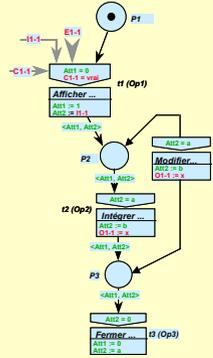


Quel type d'évaluation ?

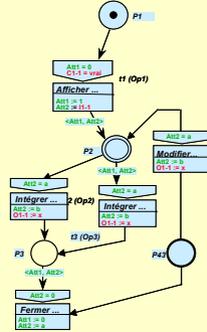


Concepteur

### Modèle de l'IHM



### Modèle de l'utilisateur (Tâche à effectuer)



itération  
Cohérence entre modèle de la tâche et le modèle IHM

B2

Propositions d'amélioration du modèle IHM

B1

Propriétés à évaluer

Cohérence du modèle de l'IHM

Propositions d'amélioration des propriétés du modèle

### Boucle B1

- Vérification de propriétés pour déterminer si la construction est correcte (ex: réinitialisable, ne présente pas de blocage,...)
- Simulation des fonctionnalités envisageables sur le modèle de l'IHM

### Boucle B2

- Vérification de performance des utilisateurs (ex: valeur prédictive de temps d'exécution, accessibilité à un état...)
- Simulation des procédures d'actions utilisateur sur l'IHM