Robots Mobiles & Autonomes

avec

Pharo

Noury Bouraqadi
http://car.mines-douai.fr/noury

JM2L – 27 nov 2010

Robots
Mobiles & Autonomes
avec
Pharo

Noury Bouraqadi – JM2L 2010

Robotique Industrielle

Première Application

Nour ouraqadi – JM2L 2010

Robots?

1. Machine, automate à l'aspect humain, capable de se **mouvoir et d'agir**. => androïde, humanoïde […]

2. Mécanisme **automatique** à commande électronique pouvant se substituer à l'homme pour effectuer certaines opérations, et capable d'en modifier de lui-même le cycle en appréhendant son environnement (=>automatique, cybernetique) […]

Le petit Robert

Robots?

Mobilité

1. Mac[illegible]ne, automate à l'aspect hur[illegible]e de se **mouvoir et d'agir**. => an[illegible]e […]

Autonomie

2. Mécanisme **automatique** à commande électronique pouvant se substituer à l'homme pour effectuer certaines opérations, et capable d'en modifier de lui-même le cycle en appréhendant son environnement (=>automatique, cybernetique) […]

Le petit Robert

Robots Mobiles & Autonomes

avec

Pharo

Robot Mobile

- **Environnement** partiellement connu et changeant
  - ➢Carte ? Lois Physiques ? Dynamique ?
- Problème de **navigation**
  - Localisation
  - Reconnaissance
  - Chemins
  - Obstacles

Robot Autonome

Décide seul

des

actions à réaliser

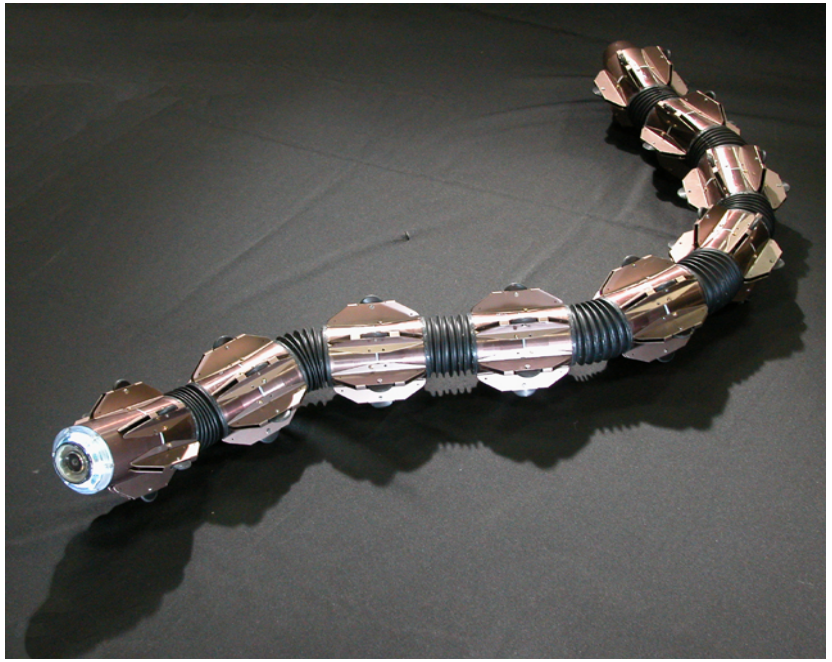
en

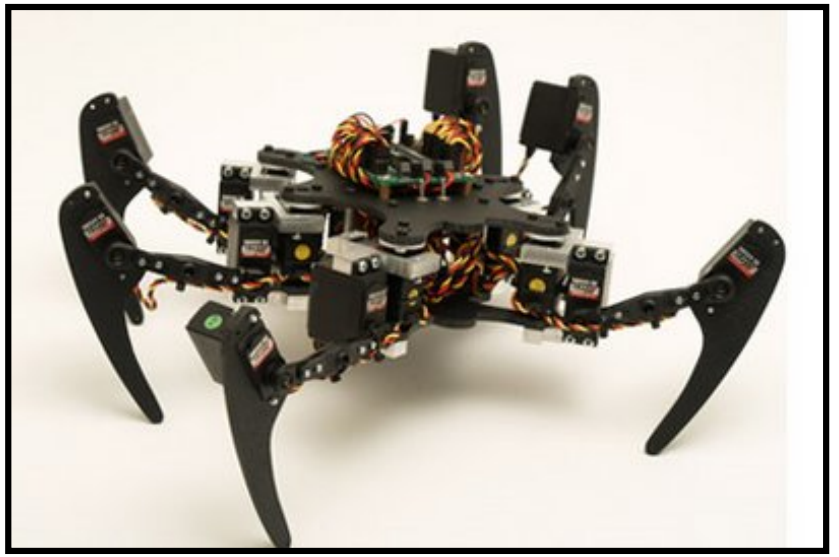
fonction de ses **perceptions**

Robotique Industrielle

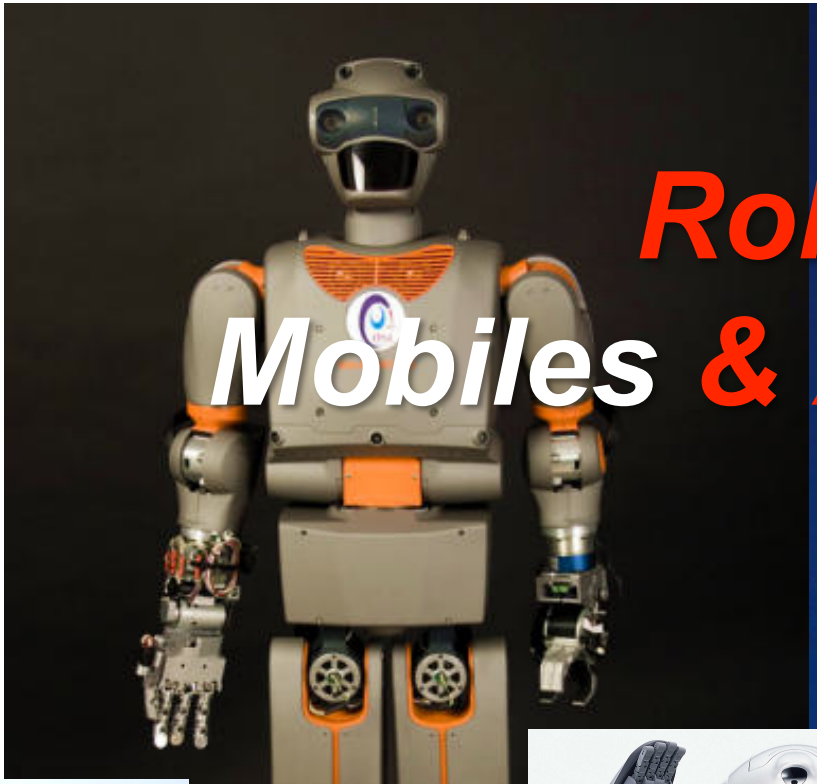
- Fonctions/trajectoires pré-définies + figées
- Pas d'interaction avec l'humain
- Environnement maîtrisé/connu

Robots Mobiles & Autonomes



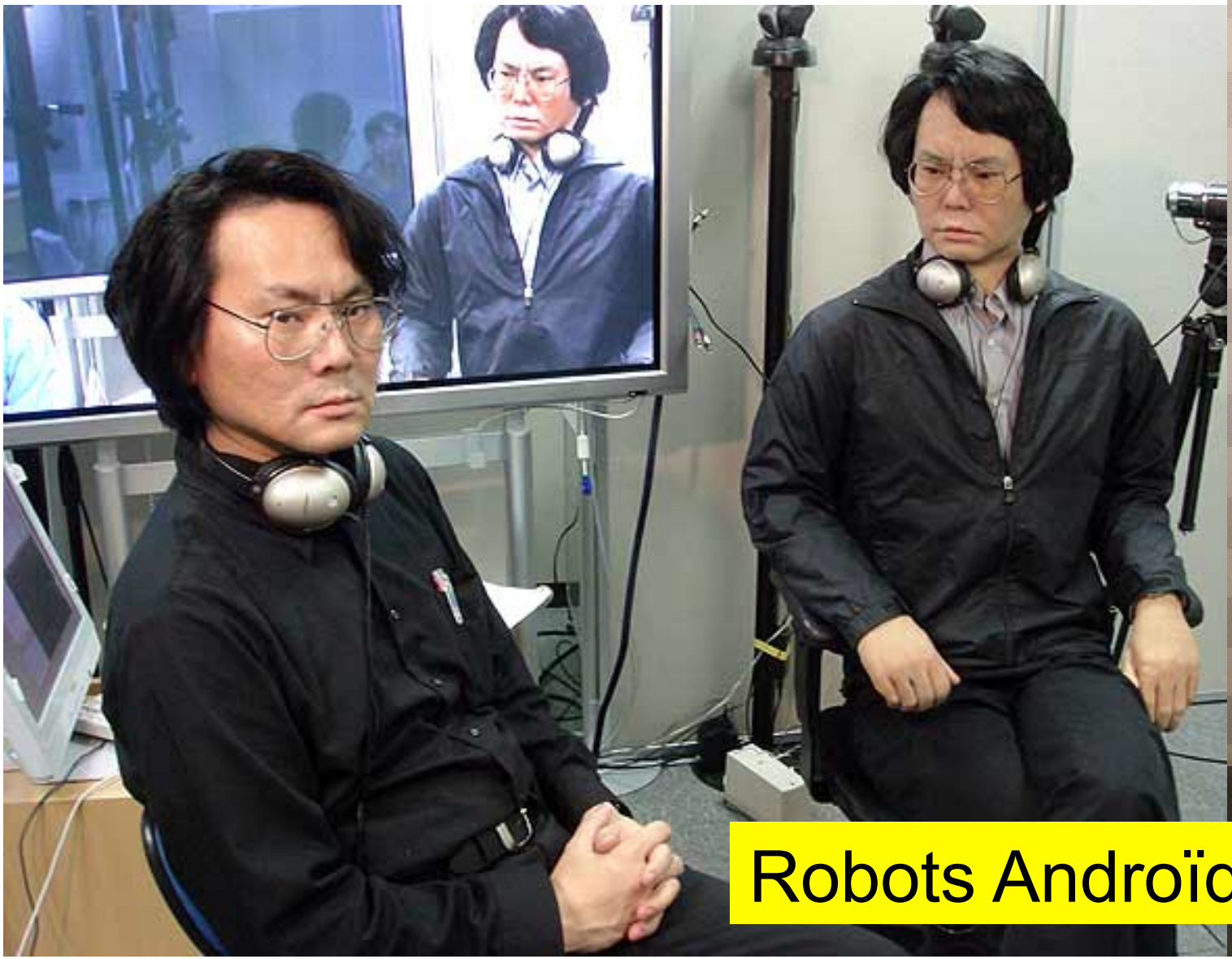


Robots Mobiles & Autonomes



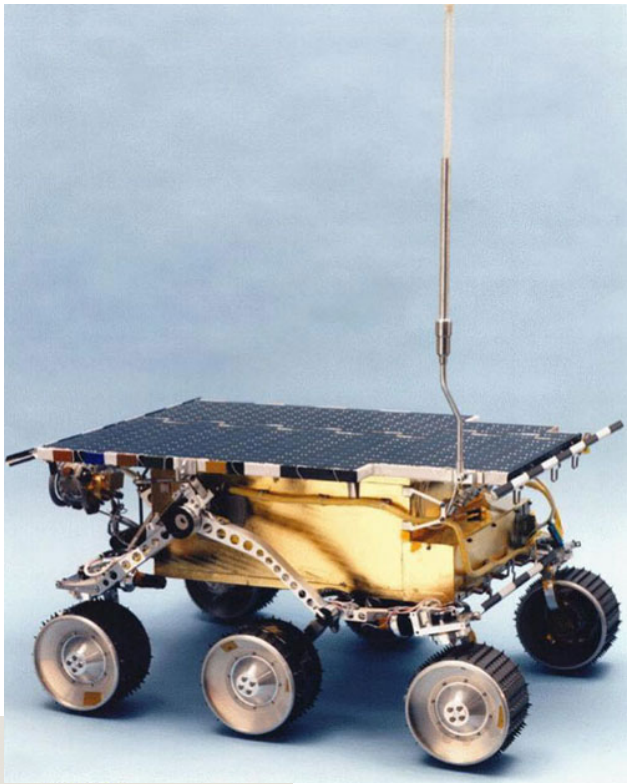
Robots Humanoïdes

Robots
Mobiles & Autonomes

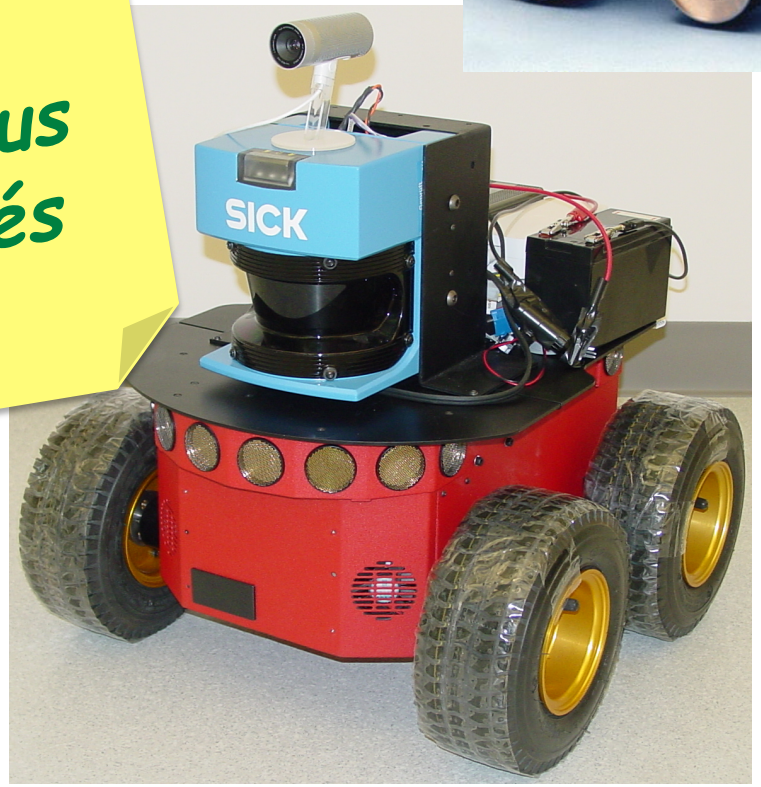


Robots Androïdes

Robots Mobiles & Autonomes



Les plus utilisés



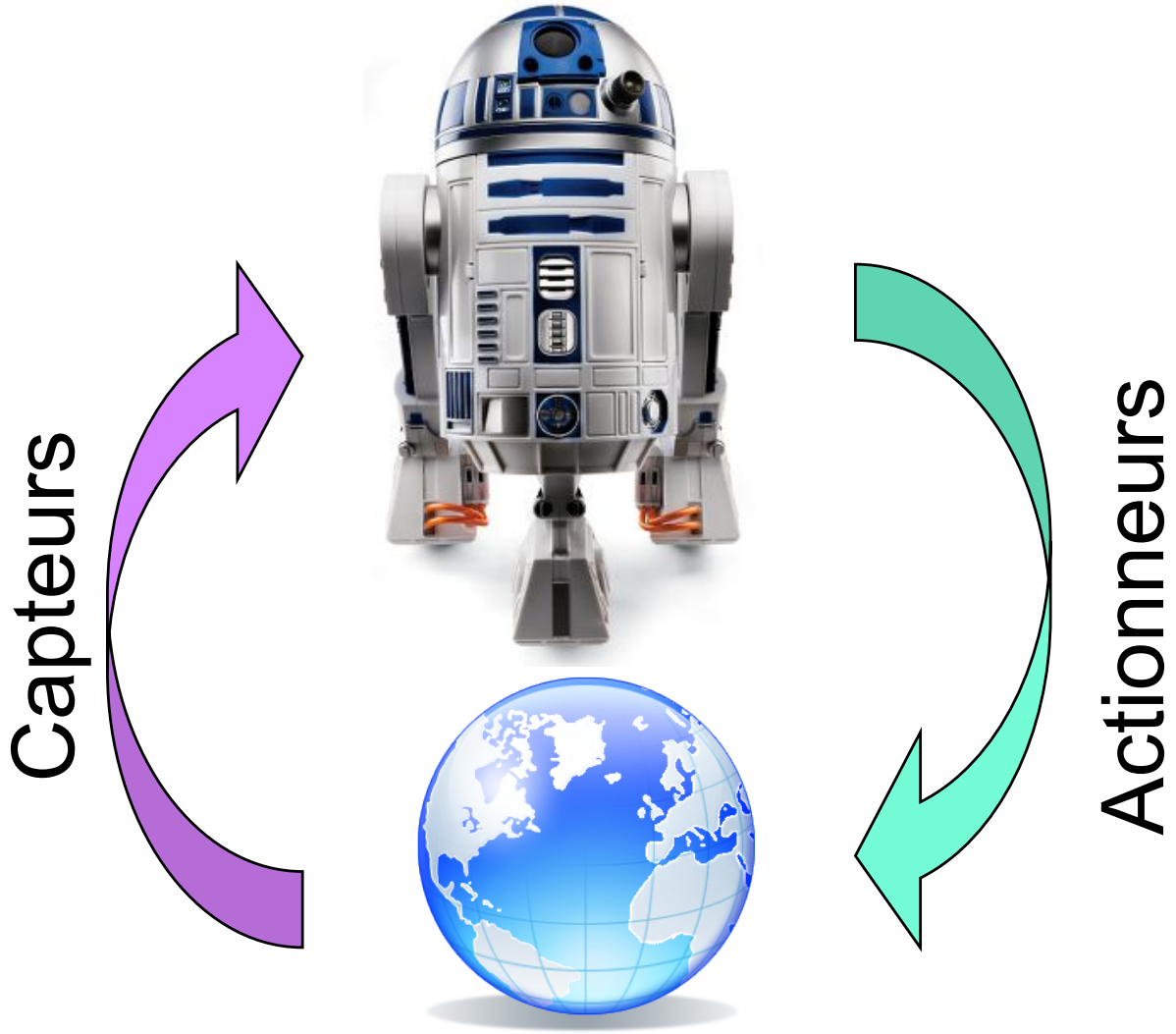
Défis

Ressources Limitées

Robots Rapides

Robots Intelligents

Logiciel de contrôle d'un robot



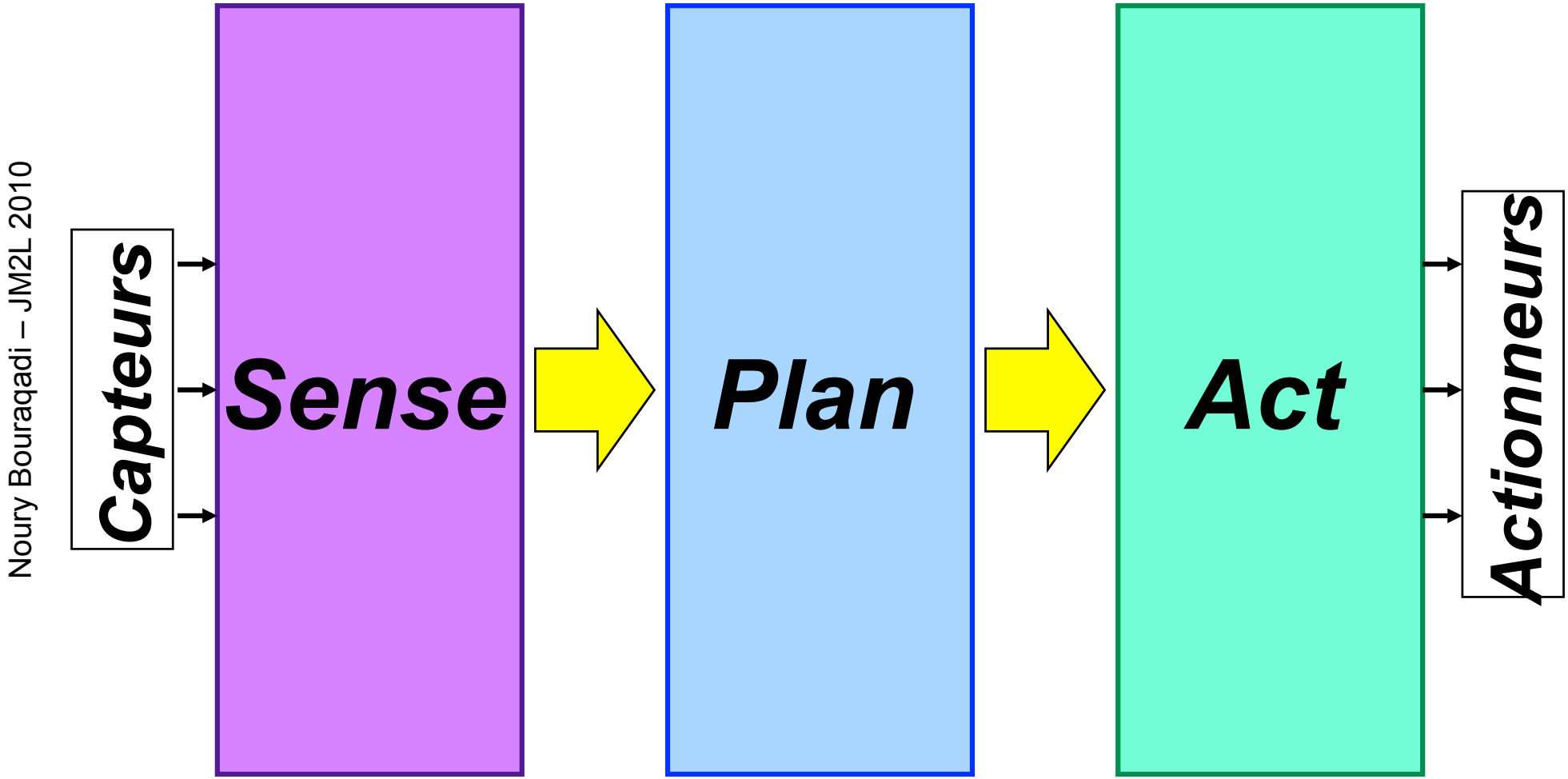
Logiciel de contrôle d'un robot

- Sense
  - Lire les capteurs
  - Extraire des informations utiles
- Plan
  - Prendre des décisions
- Act
  - Exécuter des actions simples
  - Ex: Avancer, reculer, tourner, …

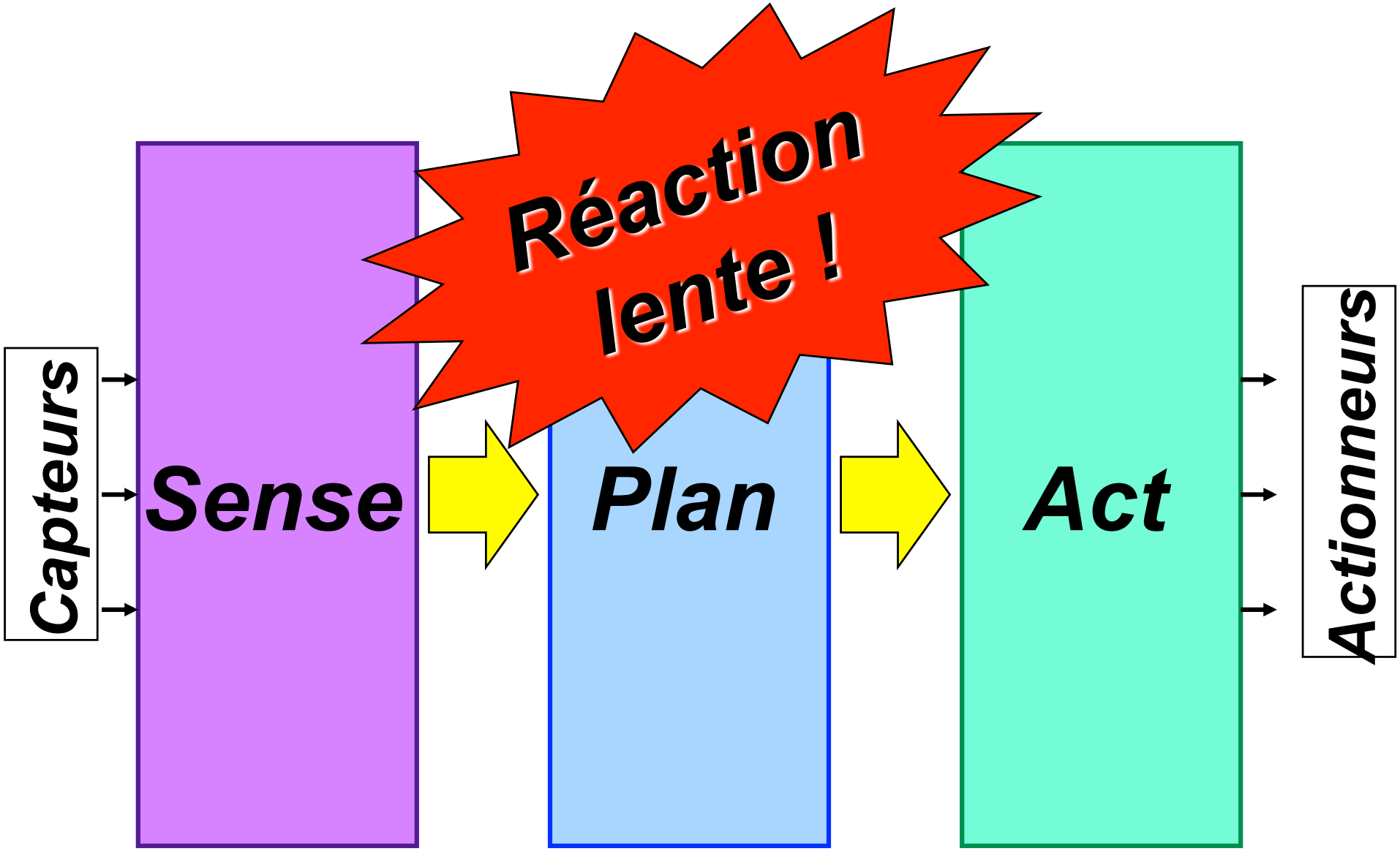
Familles d'architectures

- Délibératives
- Réactives
- Comportementales
- Hybrides

Architectures Délibératives
Think then Act



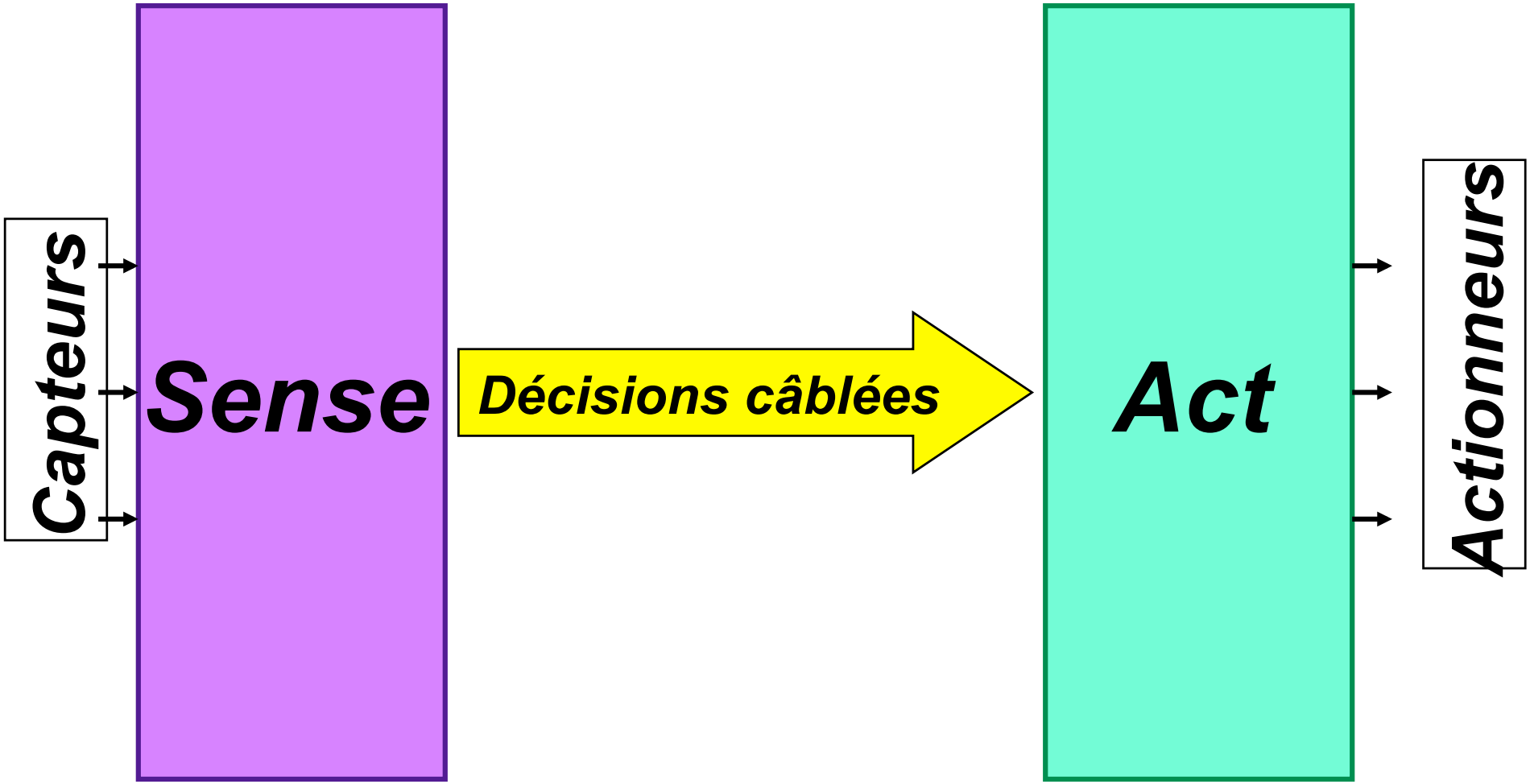
Architectures Délibératives



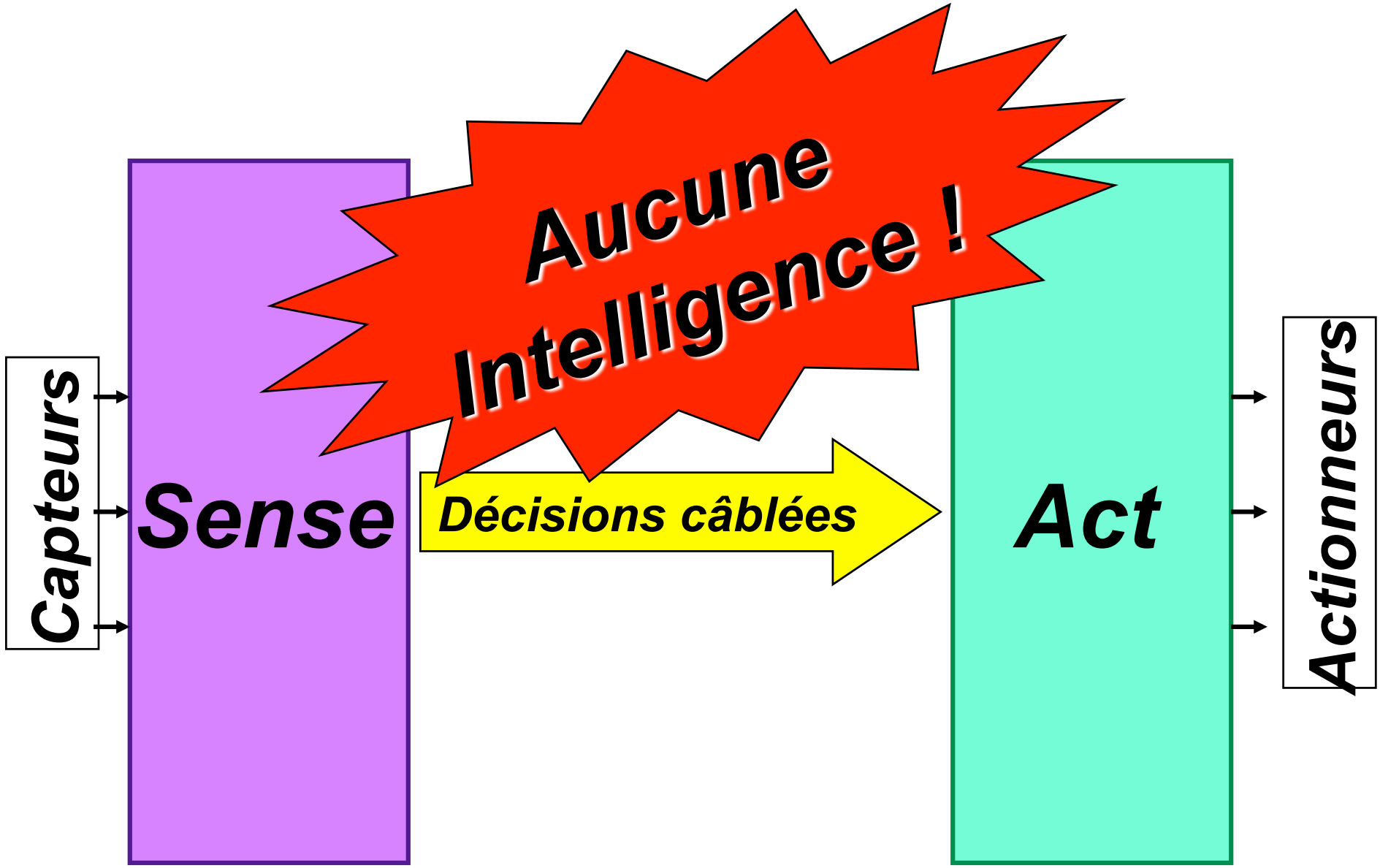
Architectures Réactives

Don't think, (Re)Act

Noury Bouraqadi - DIA - EM Douai



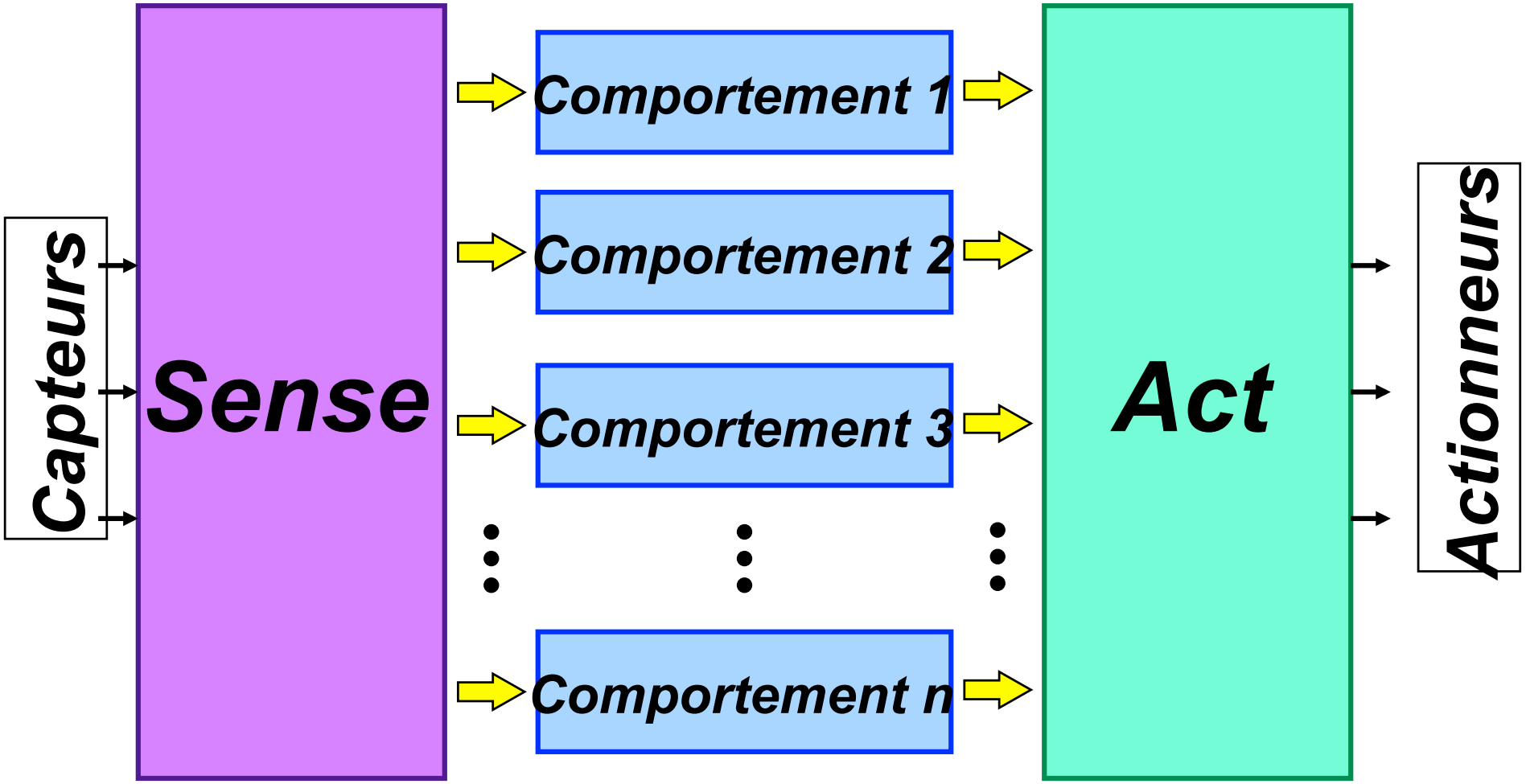
Architectures Réactives



Noury Bouraqadi – JM2L 2010

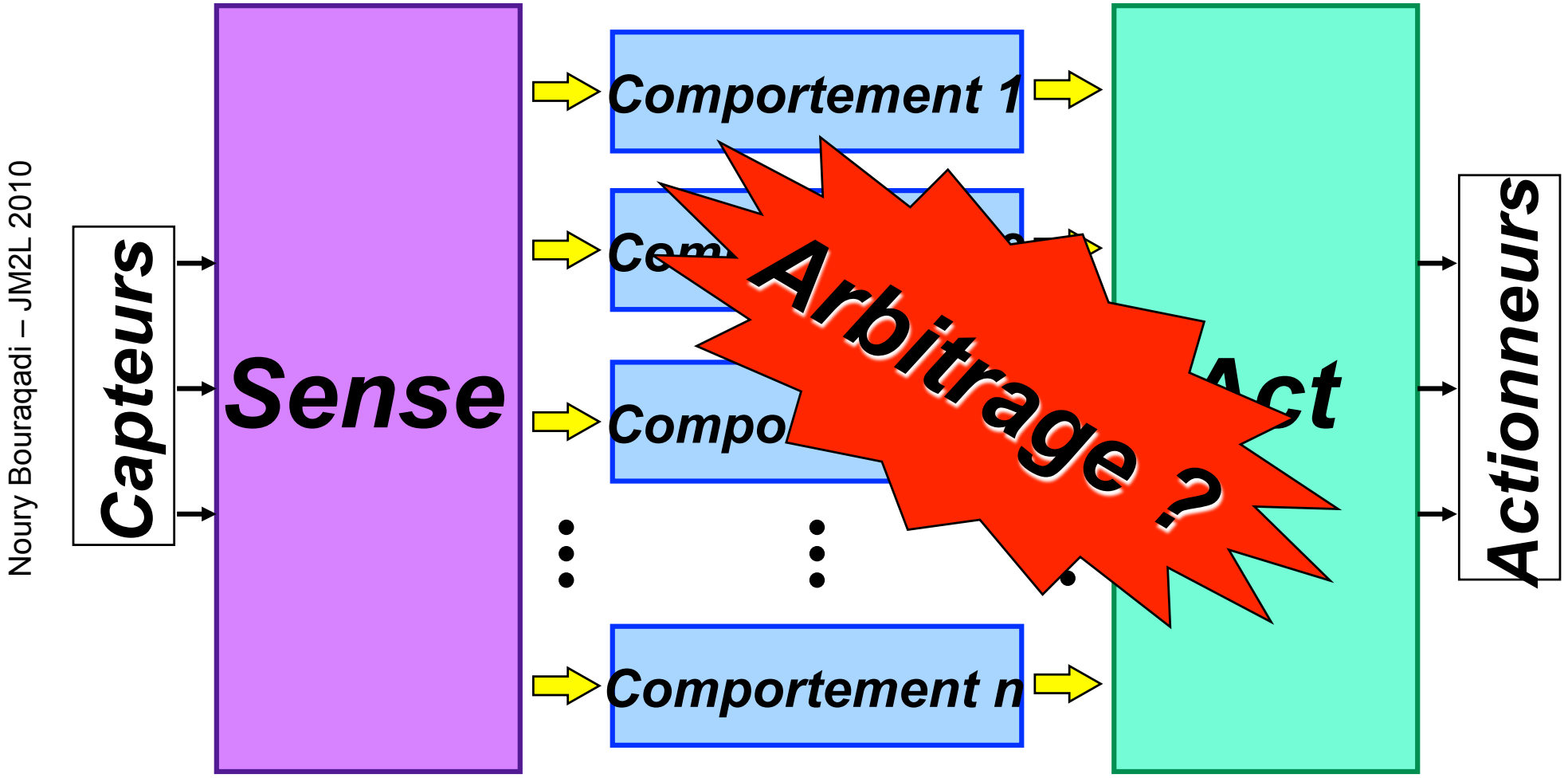
Architectures Comportementales

Think the Way You Act



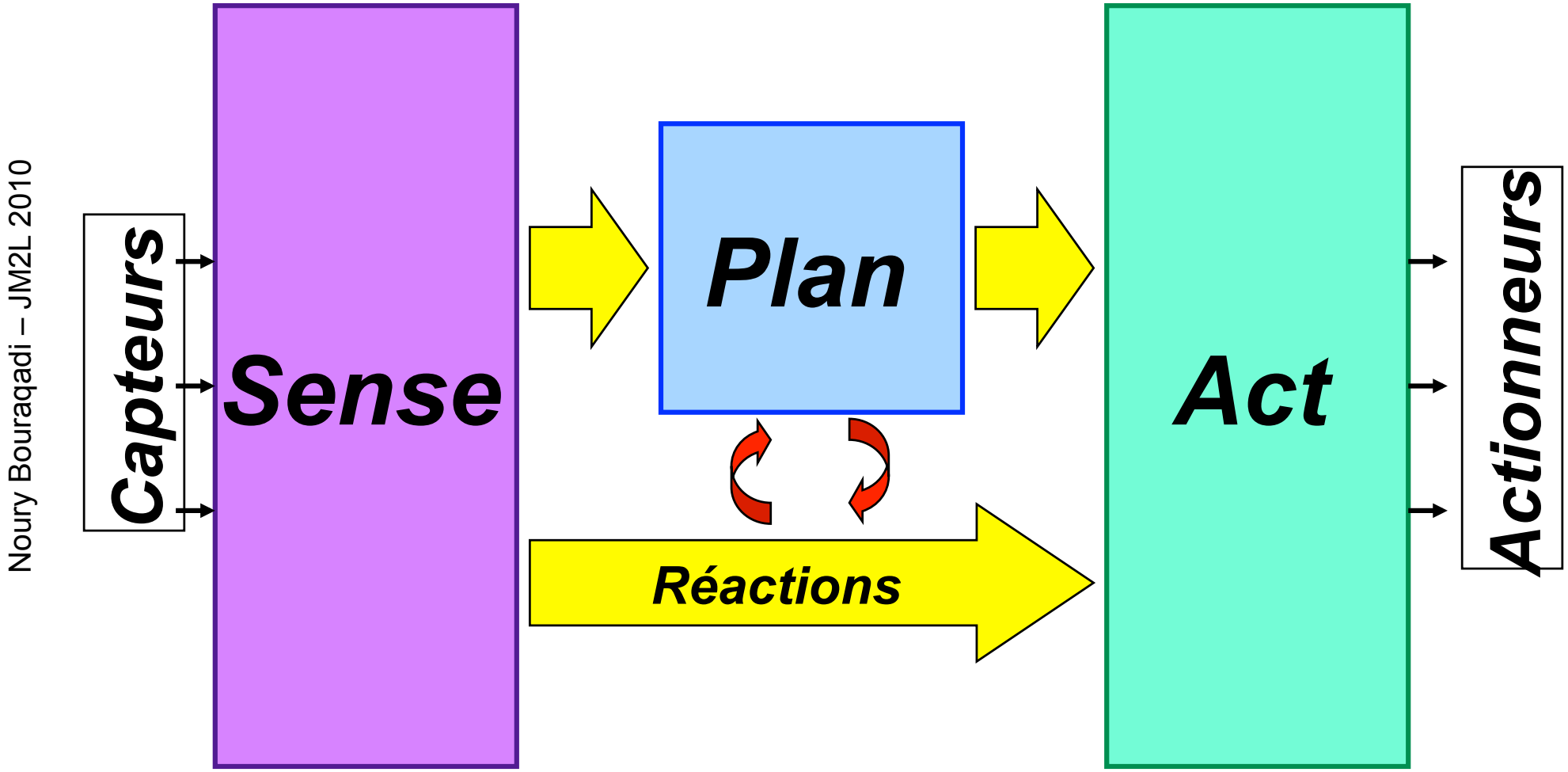
Architectures Comportementales

Think the Way You Act



Architectures Hybrides

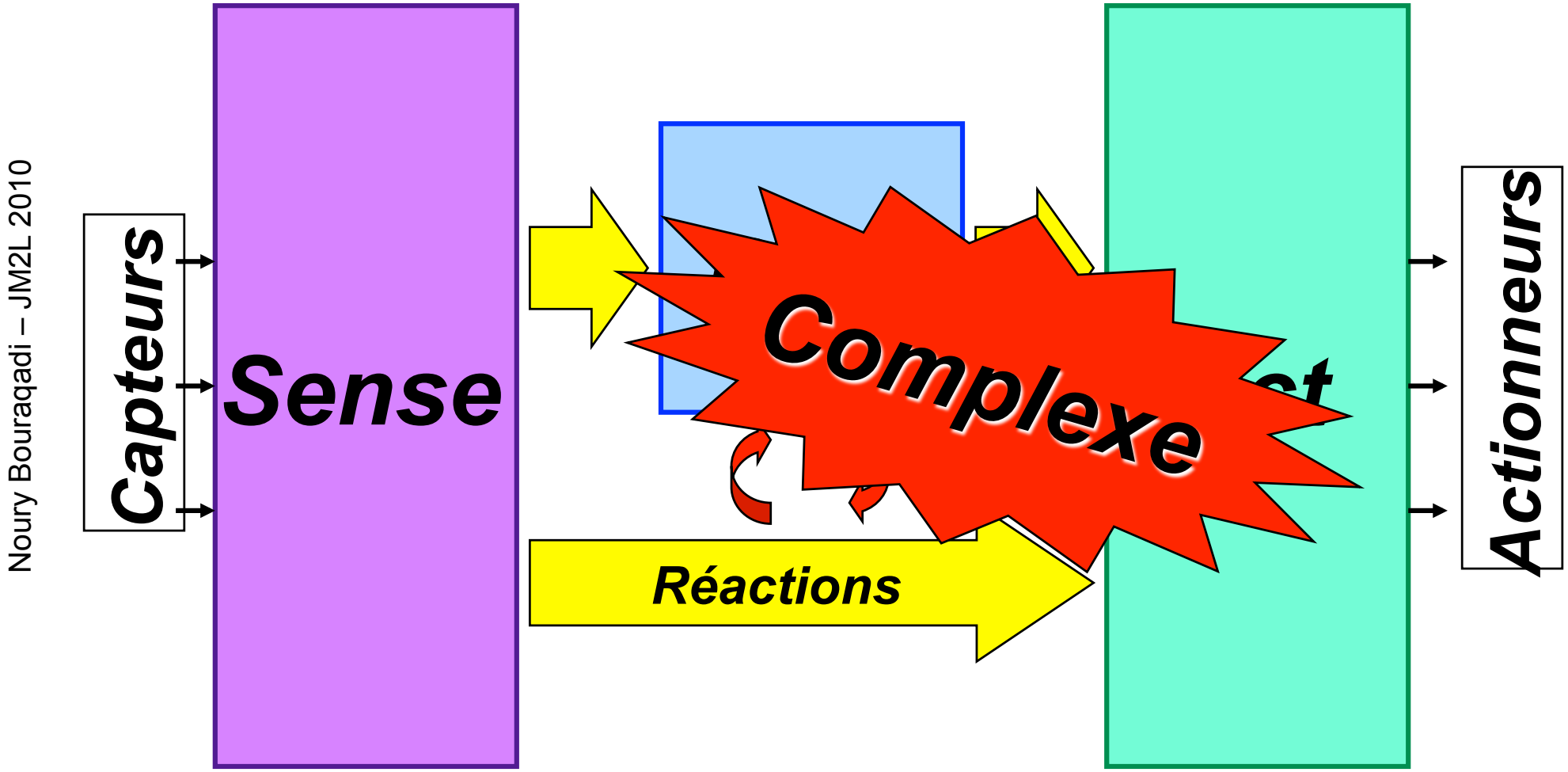
Think and Act Concurrently



Noury Bouraqadi – JM2L 2010

Architectures Hybrides

Think and Act Concurrently



Noury Bouraqadi – JM2L 2010

Robots Mobiles & Autonomes

avec

Pharo

is a
Clean
Innovative
Open-Source
Smalltalk-inspired
Environment

is a

Clean

Innovative

Open-Source

Smalltalk-inspired

Environment

Langage

+

Bibliothèques

+

Outils

is a

Clean

Innovative

Open-Source

Smalltalk-inspired

Environment

Langage dynamique à objets

is a

Clean

Innovative

Open-Source

Smalltalk-inspired

Environment

Libre !

MIT

is a

Clean

Innovative

Open-Source

Smalltalk-inspired

Environment

Traits

Relation avec la recherche

Finalité
Industrielle
&
Pédgogique

is a

Clean

Innovative

Open-Source

Smalltalk-inspired

Environment

Robots Mobiles & Autonomes

avec

Pharo

par l'exemple

Noury Bouraqadi – JM2L 2010

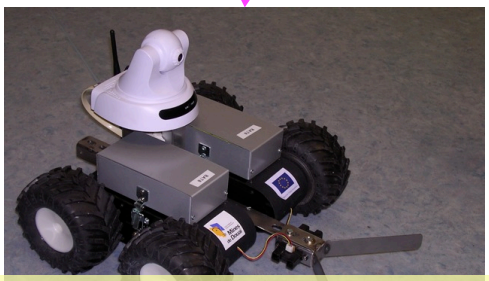
Illustration

avec

WifiBotST

Framework

pour programmer



Robots Physiques

Noury Bouraqadi – JM2L 2010

Illustration

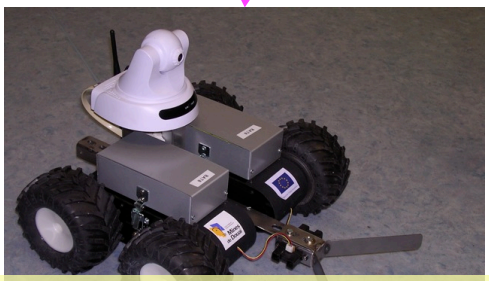
avec

WifiBotST

Développé en Pharo

Framework

pour programmer



Robots Physiques

Illustration

avec

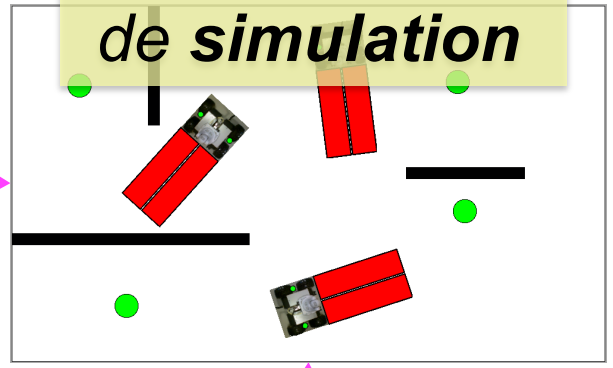
WifiBotST

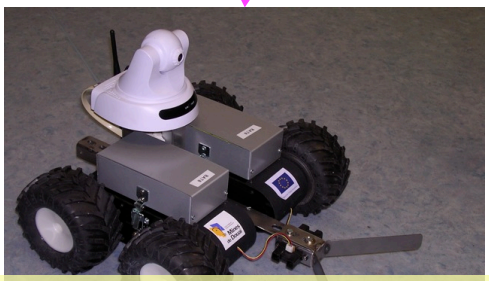
Développé en Pharo

Framework

pour programmer

*Environnement de **simulation***





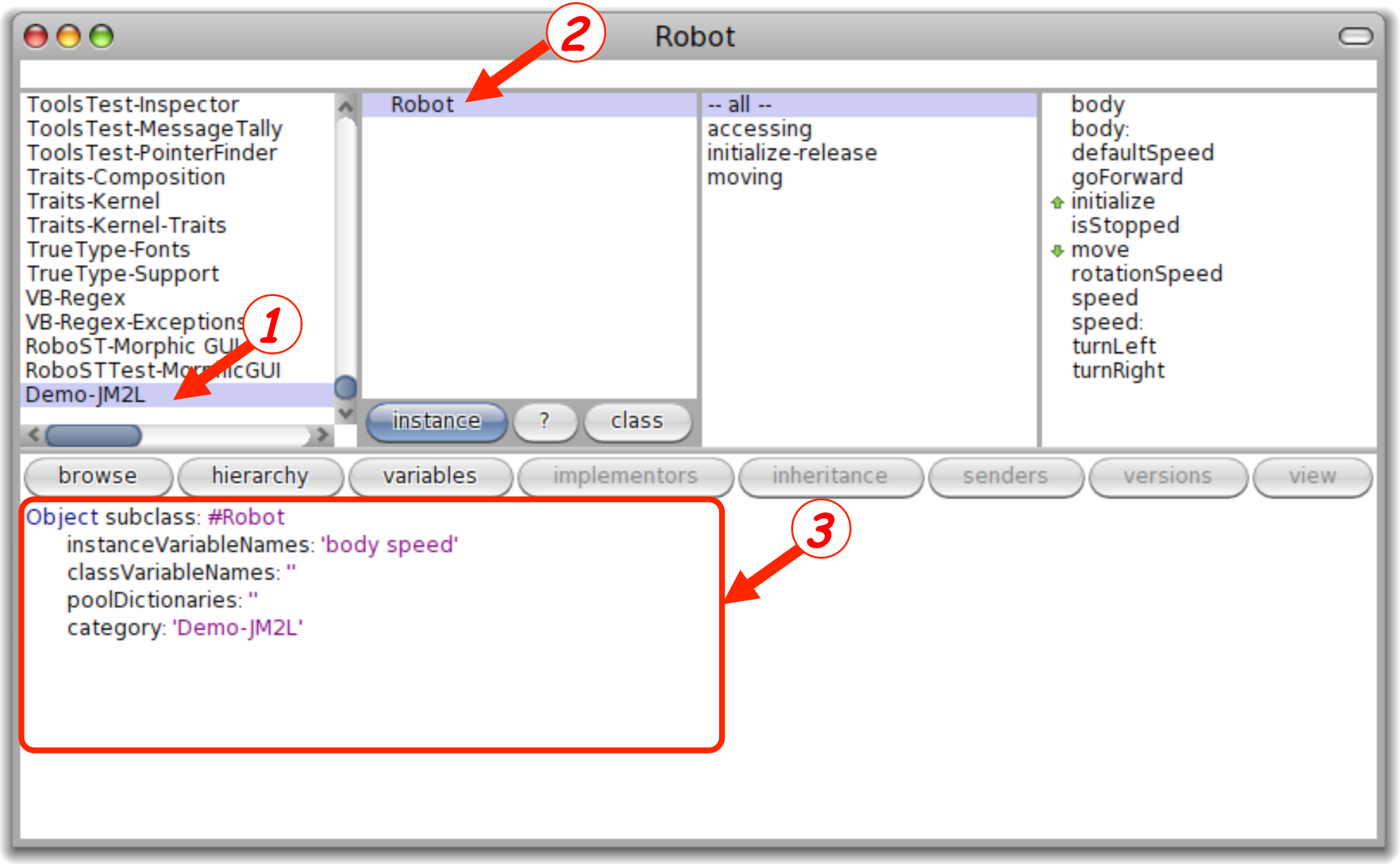
Robots Physiques

Robots Virtuels

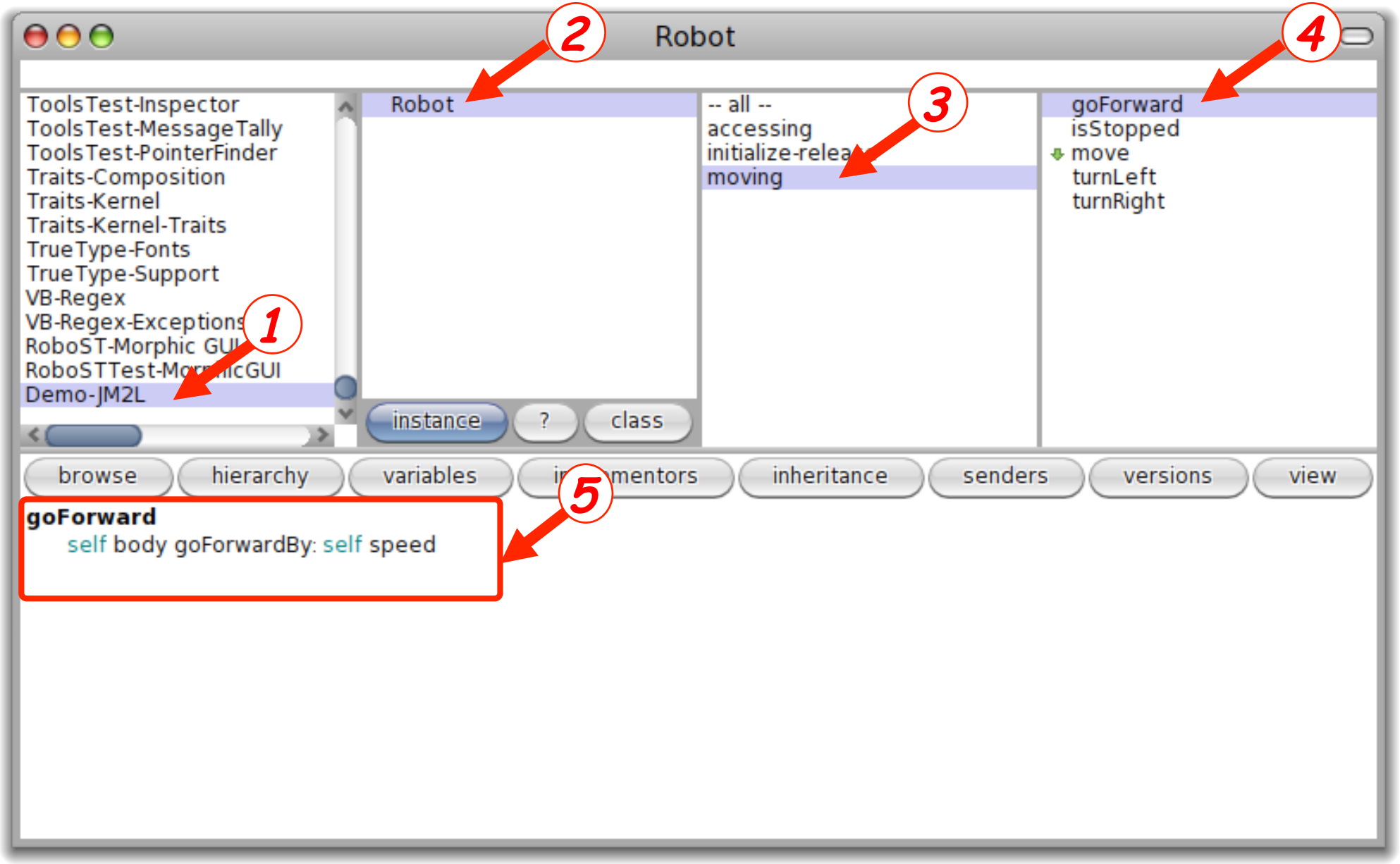
Oubliez l'archaïsme des fichiers !

Application = Ensemble d'objets

Naviguer/Editer le code



Naviguer/Editer le code



Interface avec l'OS = 4 fichiers

- Machine virtuelle
- **Image mémoire** : lecture / écriture
- Source des bibliothèques de base : lecture seule
- Sources des changements :
  - lecture / **sauvegarde automatique**
  - Quasi-impossible de perdre les sources !
  - **Versionning** sans effort !

Langage ***Simple***

- Peu de concepts / règles
  - 6 Mots réservés
  - 13 caractères spéciaux
  - 5 concepts centraux
  - 4 expressions de base
  - 1 mode de gestion de la mémoire

Langage ***Simple***

- Peu de concepts / règles
  - 6 Mots réservés
  - 13cara
  - 5 conc
  - 4 expr
  - 1 mode

- ***nil*** *objet indéfini*
  - *valeur par défaut des variables*
- ***true***, ***false*** *objets booléens*
- ***self*** *l'objet courant*
- ***super*** *l'objet courant*

dans le contexte de la superclasse

- ***thisContext*** *partie de la pile d'exécution*

correspondant à la méthode courante

Langage ***Simple***

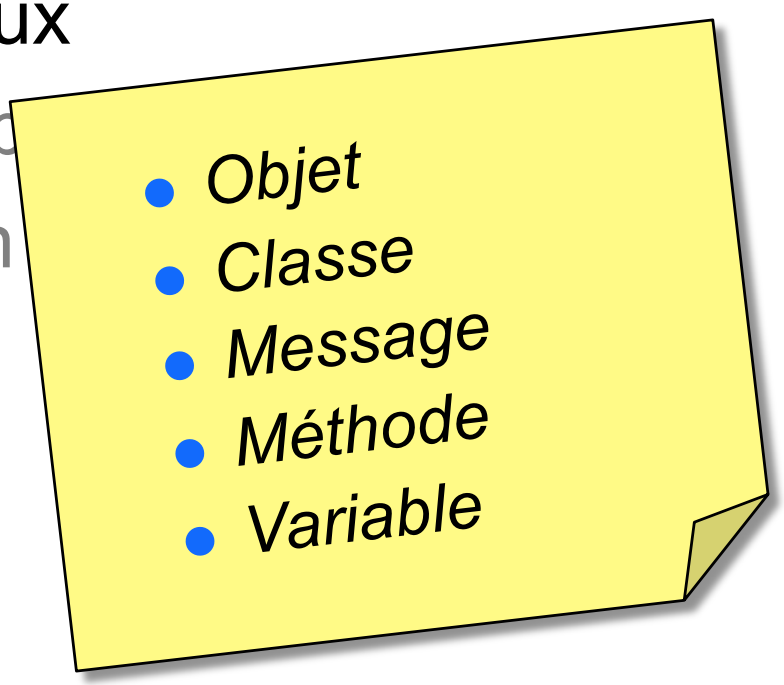
- Peu de concepts / règles
  - 6 Mots réservés
  - 13 caractères spéciaux

- *:= affectation*
- *^ return*
- *| variable1 variable2 |*
- *$ caractère*
- *# littéral*
- *{ } tableau*

- *. termine toute expression*
- *; cascade de messages*
- *( ) précédence*
- *[ ] bloc de code*
- *" commentaire"*
- *<pragma>*
- *'chaîne de caractères'*

Langage ***Simple***

- Peu de concepts / règles
  - 6 Mots réservés
  - 13 caractères spéciaux
  - 5 concepts centraux
  - 4 expressions de b
  - 1 mode de gestion



Langage ***Simple***

- Peu de concepts / règles
  - 6 Mots réservés
  - 13 caractères spéciaux
  - 5 concepts centraux
  - 4 expressions de base
  - 1 mode de gesti

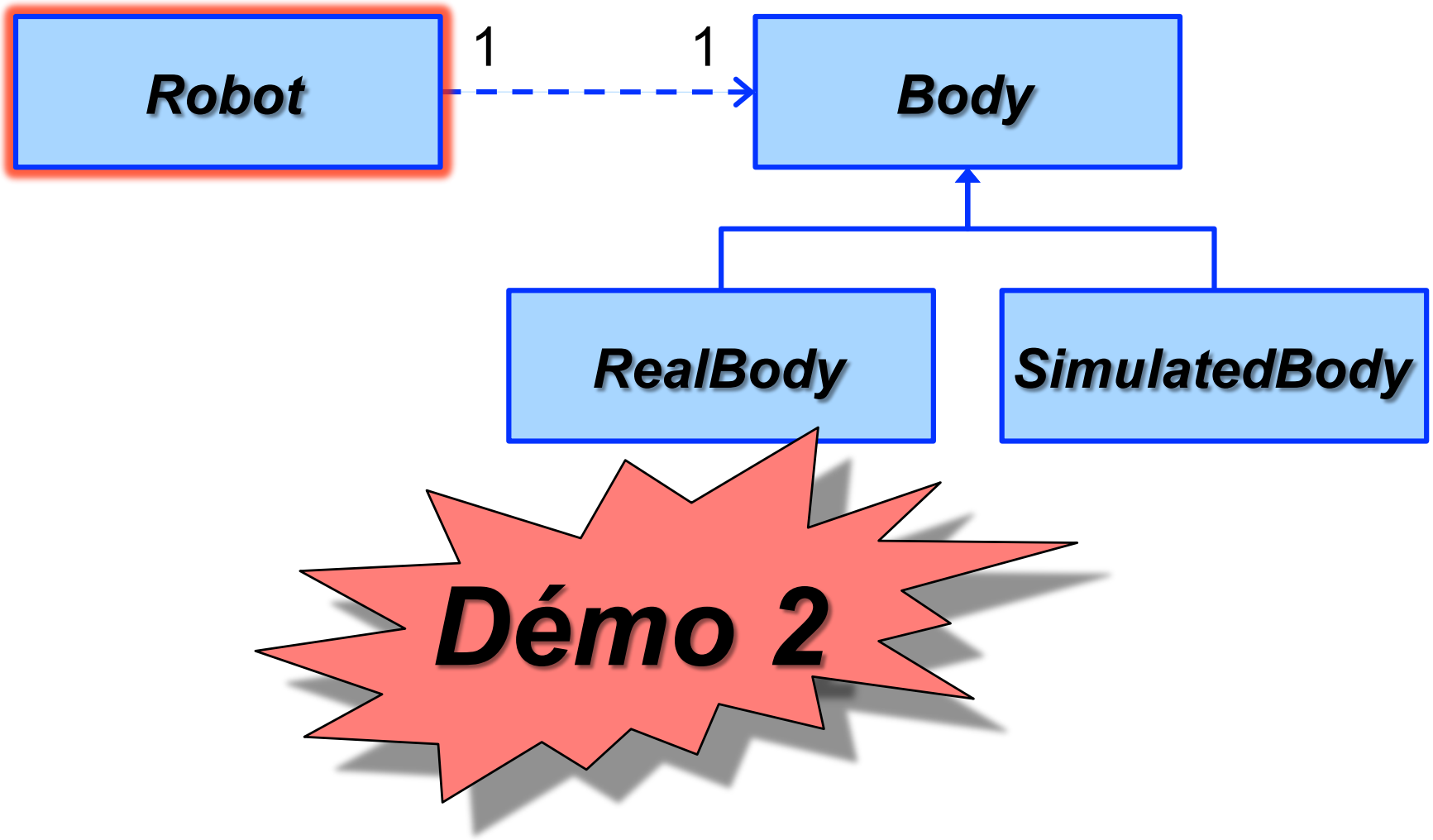
- *Déclaration d'une variable*
- *Affectation d'une variable*
- *Lecture d'une variable*
- *Envoi d'un message*

Langage ***Simple***

- Peu de concepts / règles
  - 6 Mots réservés
  - 13 caractères spéciaux
  - 5 concepts centraux
  - 4 expressions de base
  - 1 mode de gestion de la mémoire

Ramasse Miettes

***Automatique** utilise un "Garbage Collector"*


Robot
1
1
Body
RealBody
SimulatedBody
Démo 2

Langage ***Dynamique***

Tout se passe à **l'exécution**

- Définition des classes
- Compilation
- Vérification de type
- …

Langage ***Dynamique***

Tout se passe à **l'exécution**

- Définition des classes
- Compilation
- Vérification de type
- …

Langage *Dynamique*

Tout se passe à **l'exécution**

- Définition des classes
- Compilation
- Vérification de type
- …

Langage *Dynamique*

Tout se passe à **l'exécution**

- Définition des classes
- Compilation
- Vérification de type
- …

Langage dynamique !

Modification de la classe **à l'exécution**

Après la création d'instances

Langage ***Uniforme***

- Pas d'exception aux règles
  - **Tout** est objet
    - nombres, tableaux, classes, compilateur, IDE, …
  - **Toute** méthode retourne une valeur
  - **Toute** action == envoi de message
    - création d'objets, définition de classes, if, while, …
  - **Toutes** les classes et les méthodes sont publics
  - **Tous** les champs sont privés
  - **Tout** objet est instance d'une classe

Langage ***Uniforme***

- Pas d'exception aux règles
  - **Tout** est objet
    - nombres, tableaux, classes, compilateur, IDE, …
  - **Toute** méthode retourne une valeur
  - **Toute** action == envoi de message
    - création d'objets, définition de classes, if, while, …
  - **Toutes** les classes et les méthodes sont publics
  - **Tous** les champs sont privés
  - **Tout** objet est instance d'une classe

Itérations & Blocs de code

- Répéter un nombre de fois
- Répéter tant que

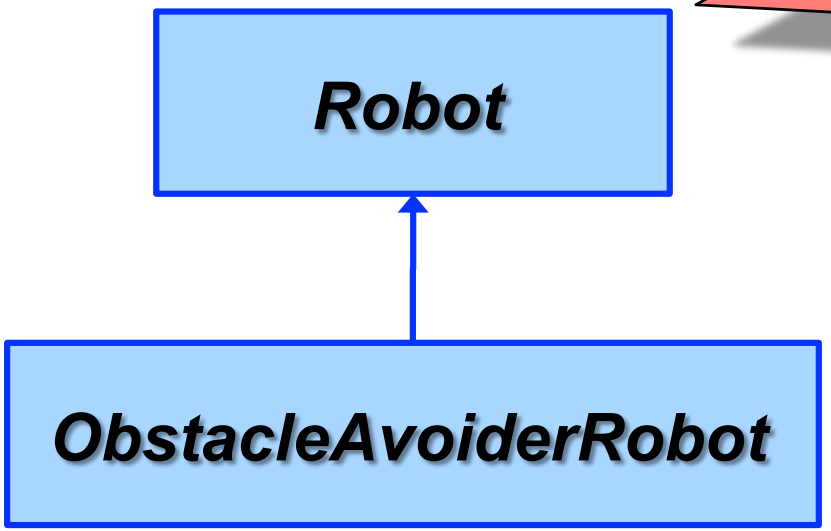
Langage ***Uniforme***

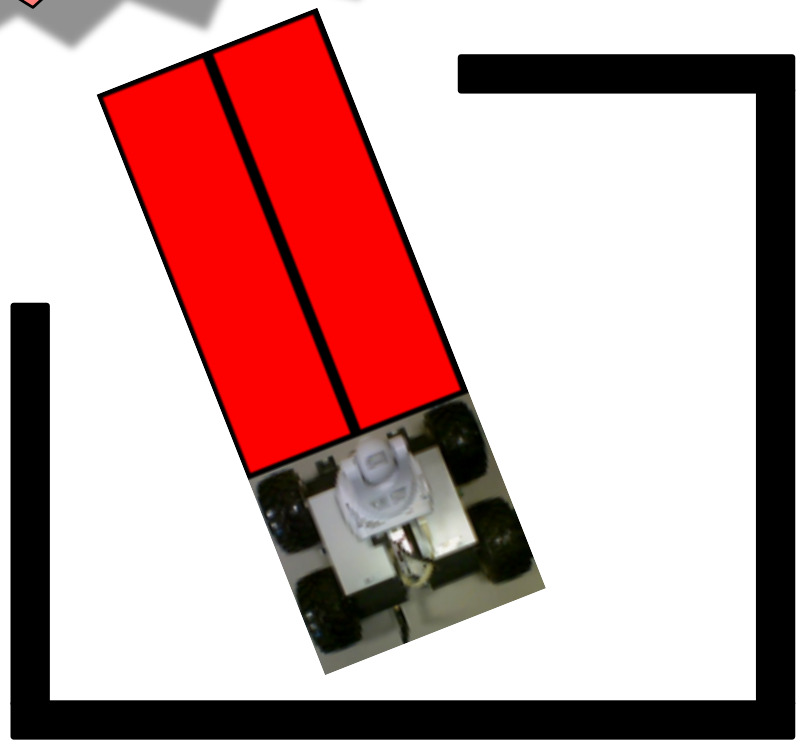
- Pas d'exception aux règles
  - **Tout** est objet
    - nombres, tableaux, classes, compilateur, IDE, …
  - **Toute** méthode retourne une valeur
  - **Toute** action == envoi de message
    - création d'objets, définition de classes, if, while, …
  - **Toutes** les classes et les méthodes sont publics
  - **Tous** les champs sont privés
  - **Tout** objet est instance d'une classe

Langage ***Uniforme***

- Pas d'exception aux règles
  - **Tout** est objet
    - nombres, tableaux, classes, compilateur, IDE, …
  - **Toute** méthode retourne une valeur
  - **Toute** action == envoi de message
    - création d'objets, définition de classes, if, while, …
  - **Toutes** les classes et les méthodes sont publics
  - **Tous** les champs sont privés
  - **Tout** objet est instance d'une classe

Héritage & Conditionnelles





Langage ***Uniforme***

- Pas d'exception aux règles
  - **Tout** est objet
    - nombres, tableaux, classes, compilateur, IDE, …
  - **Toute** méthode retourne une valeur
  - **Toute** action == envoi de message
    - création d'objets, définition de classes, if, while, …
  - **Toutes** les classes et les méthodes sont publics
  - **Tous** les champs sont privés
  - **Tout** objet est instance d'une classe

Langage ***Uniforme***

- Pas d'exception aux règles
  - **Tout** est objet
    - nombres, tableau… IDE, …
  - **Toute** méthod…
  - **Toute** action =…
    - création d'objets, …ses, if, while, …
  - **Toutes** les classes et les méthodes sont publics
  - **Tous** les champs sont privés
  - **Tout** objet est instance d'une classe

Variables accessibles par un seul objet

Langage ***Uniforme***

- Pas d'exception aux règles
  - **Tout** est objet
    - nombres, tableaux, classes, compilateur, IDE, …
  - **Toute** méthode retourne une valeur
  - **Toute** action == envoi de message
    - création d'objets, définition de classes, if, while, …
  - **Toutes** les classes et les méthodes sont publics
  - **Tous** les champs sont privés
  - **Tout** objet est instance d'une classe

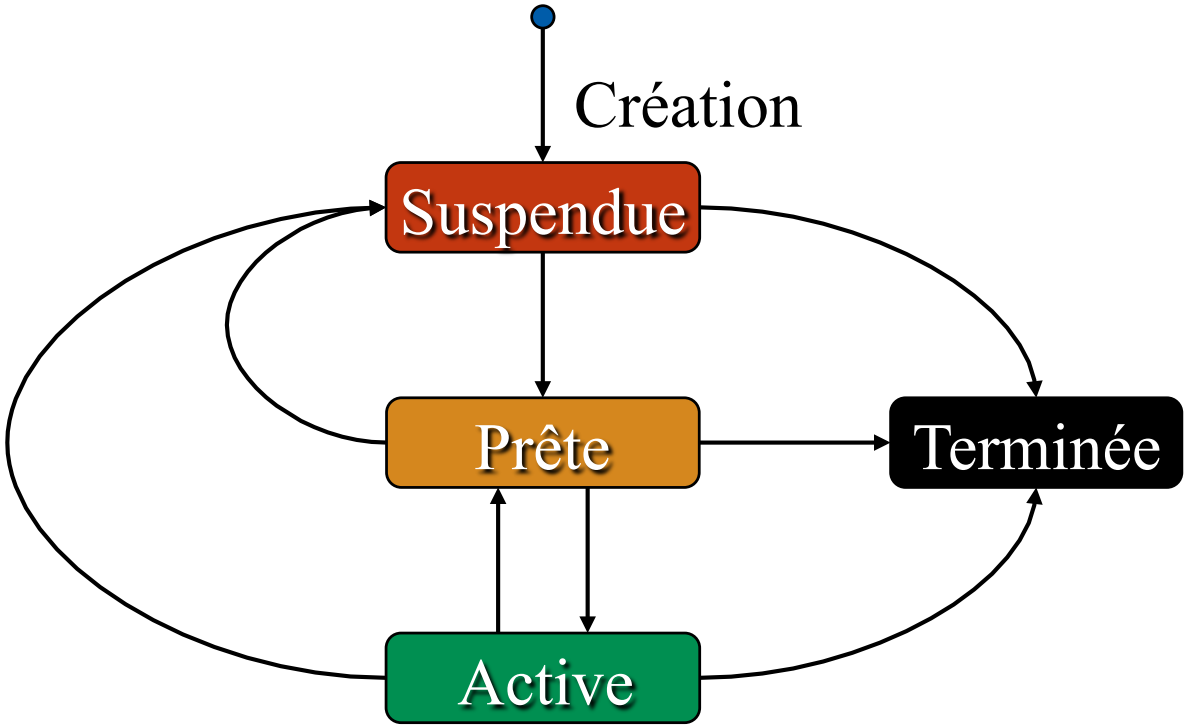
Les classes sont des objets !

Démo 6

Les procesus sont des objets

Multi-tâche préemptif

- Chaque tâche a une priorité **modifiable**
- Exécution de la tâche prête la plus prioritaire

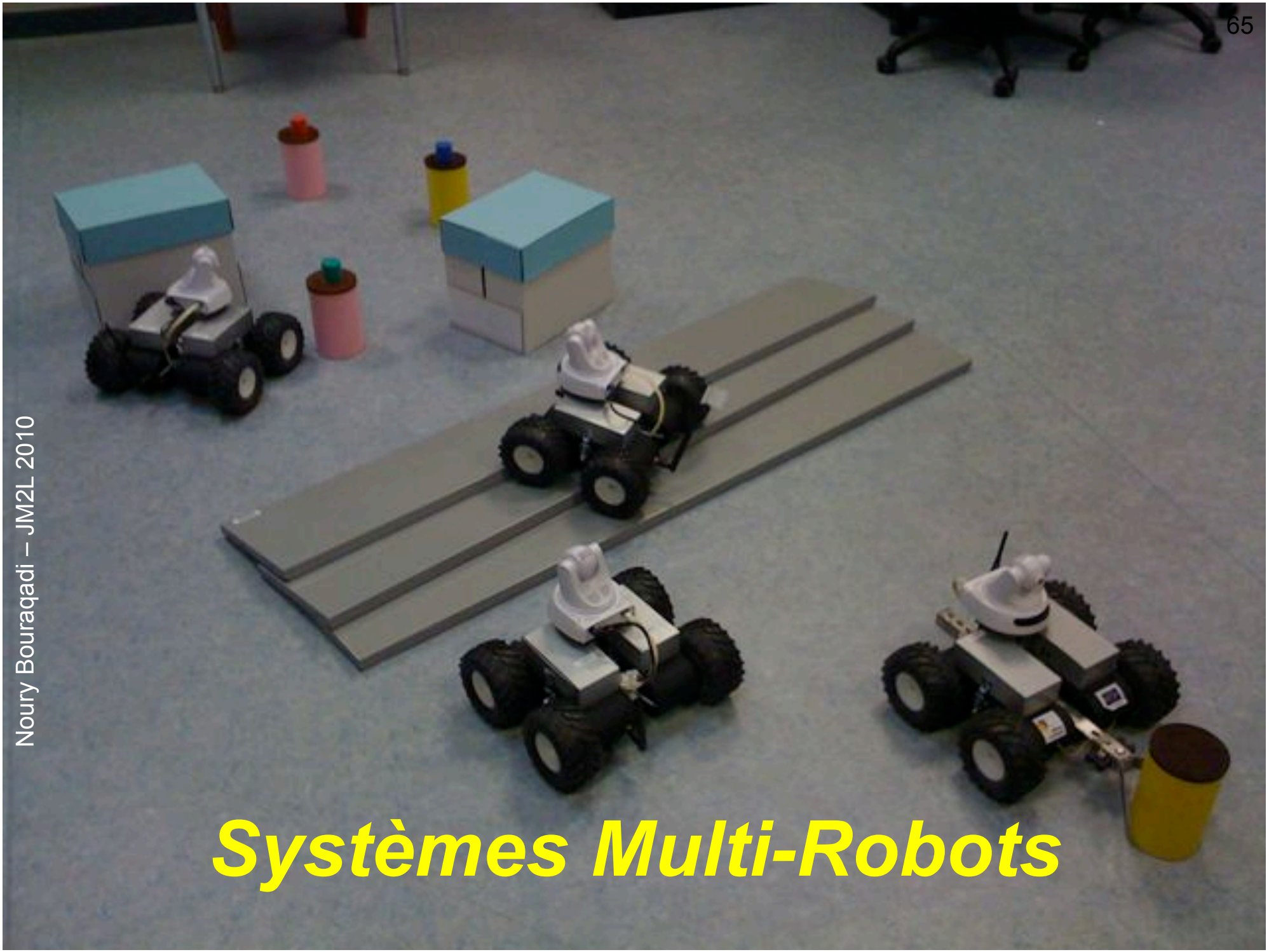


Les procesus sont des objets

Multi-tâche préemptif

- Chaque tâche a une priorité **modifiable**
- Exécution de la tâche prête la plus prioritaire

Démo 7



Systèmes Multi-Robots

Noury Bouraqadi – JM2L 2010

Collections & itérations

- Bibliothèque de collections très riche
- Itérer sur une collection = envoi de message

Langage ***Puissant***

Réflexion

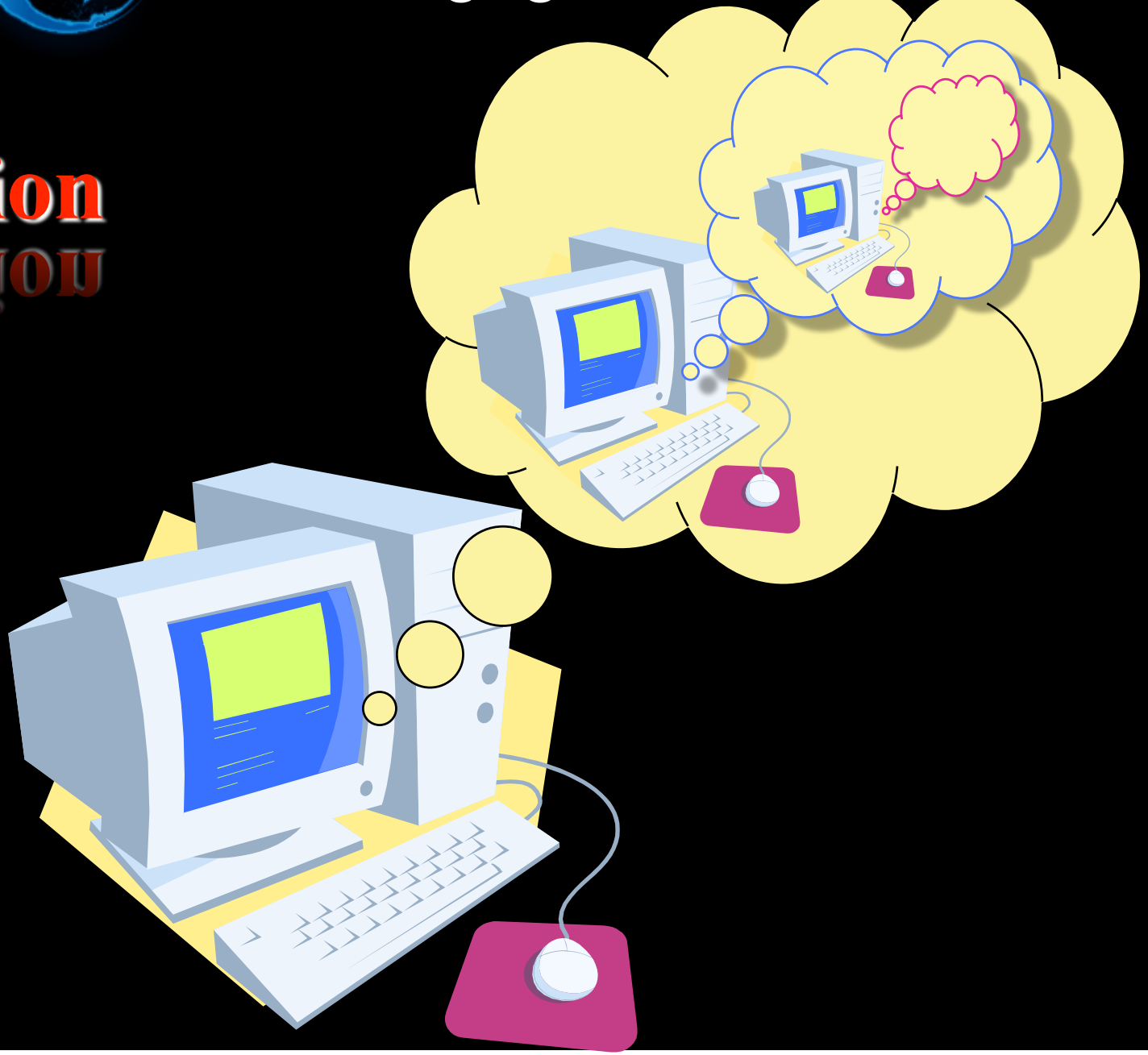
Réflexion

Noury Bouraqadi – JM2L 2010

Pharo

Langage ***Puissant***

Réflexion



Noury Bouraqadi – JM2L 2010

Langage *Puissant*

Réflexion

Classes

Objets représentant :

- *Entités du langage*
- *Environnement d'exécution*
- *Outils de développement*

Messages

Blocs

Méthodes

Noury Bouraqadi – JM2L 2010

Langage *Puissant*

Réflexion

Ordonnanceur

Objets représentant :

- *Entités du langage*
- *Environnement d'exécution*
- *Outils de développement*

Pile d'exécution

Processus

Noury Bouraqadi – JM2L 2010

Langage *Puissant*

Réflexion

Compilateur

Objets représentant :

- *Entités du langage*
- *Environnement d'exécution*
- *Outils de développement*

Debugger

Browser

Console

Noury Bouraqadi – JM2L 2010

Langage ***Puissant***

Réflexion

Toutes les classes sont

modifiables

Langage *Puissant*

Réflexion

Toutes les classes sont ***modifiables***

Object

Metaclass

Class

Process

Message

Compiler

UndefinedObject

Debugger

Browser

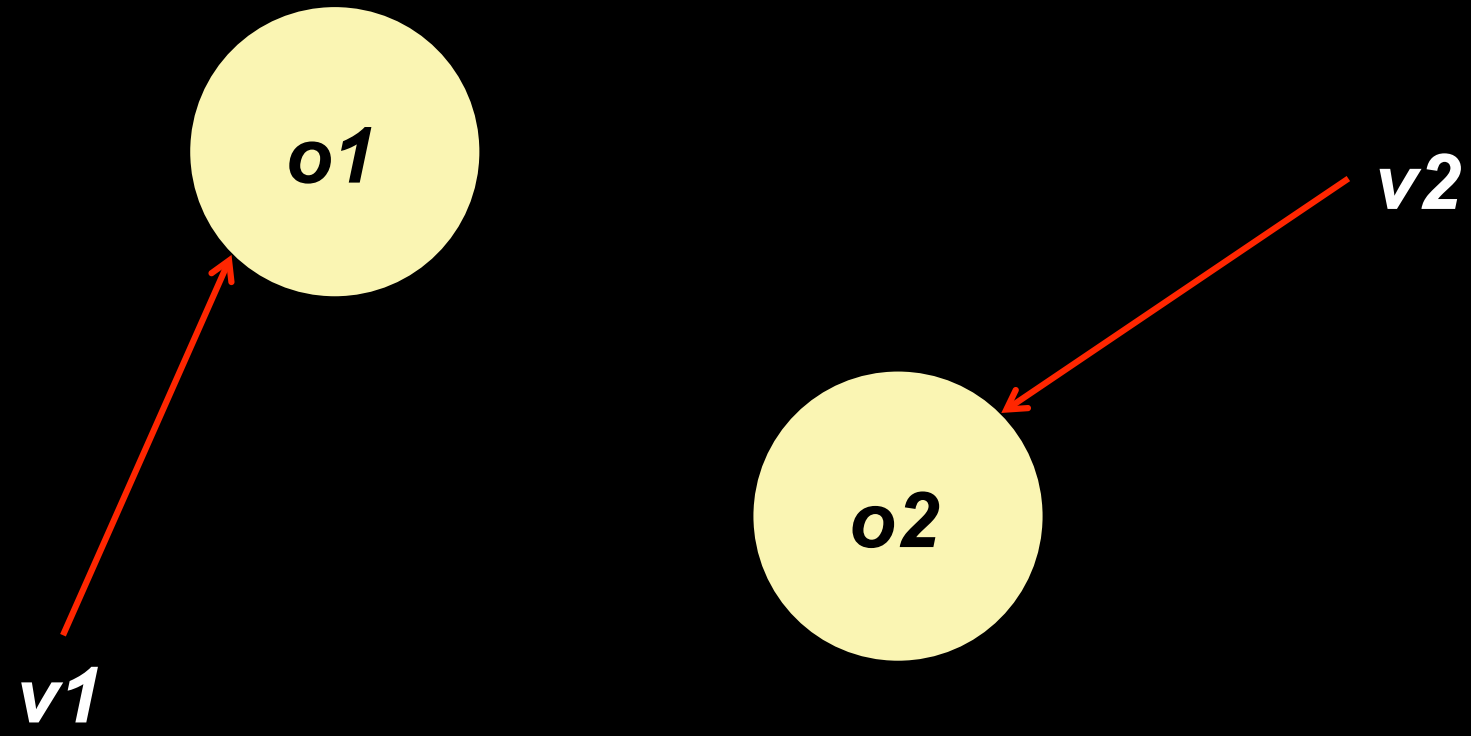
BlockClosure

Boolean

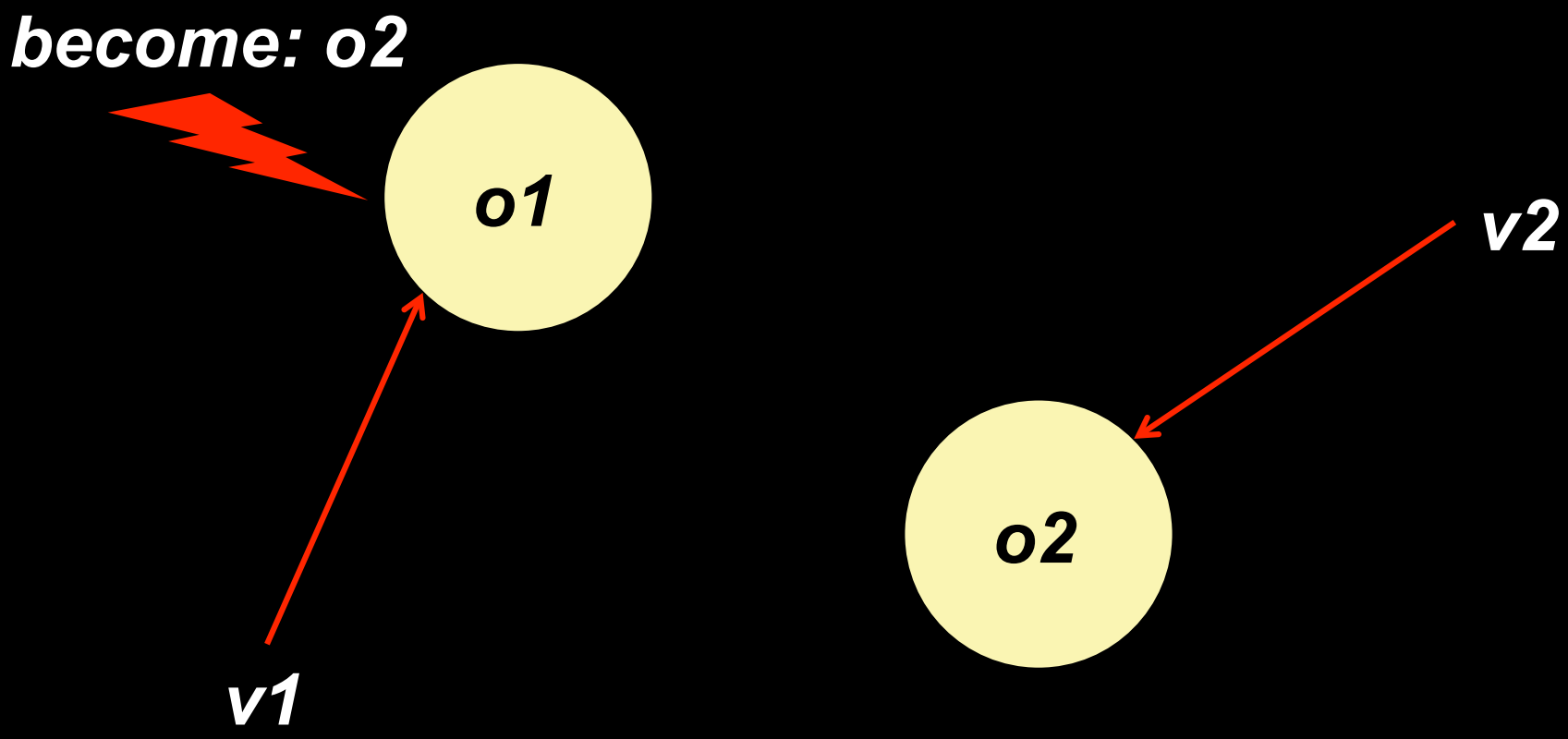
Noury Bouraqadi – JM2L 2010


Pharo
Langage Puissant
Réflexion
Retrouver les références sur un objet
Changer la classe d'un objet
Echanger l'identité de 2 objets
Méta-Opérations
Noury Bouraqadi IM2L 2010

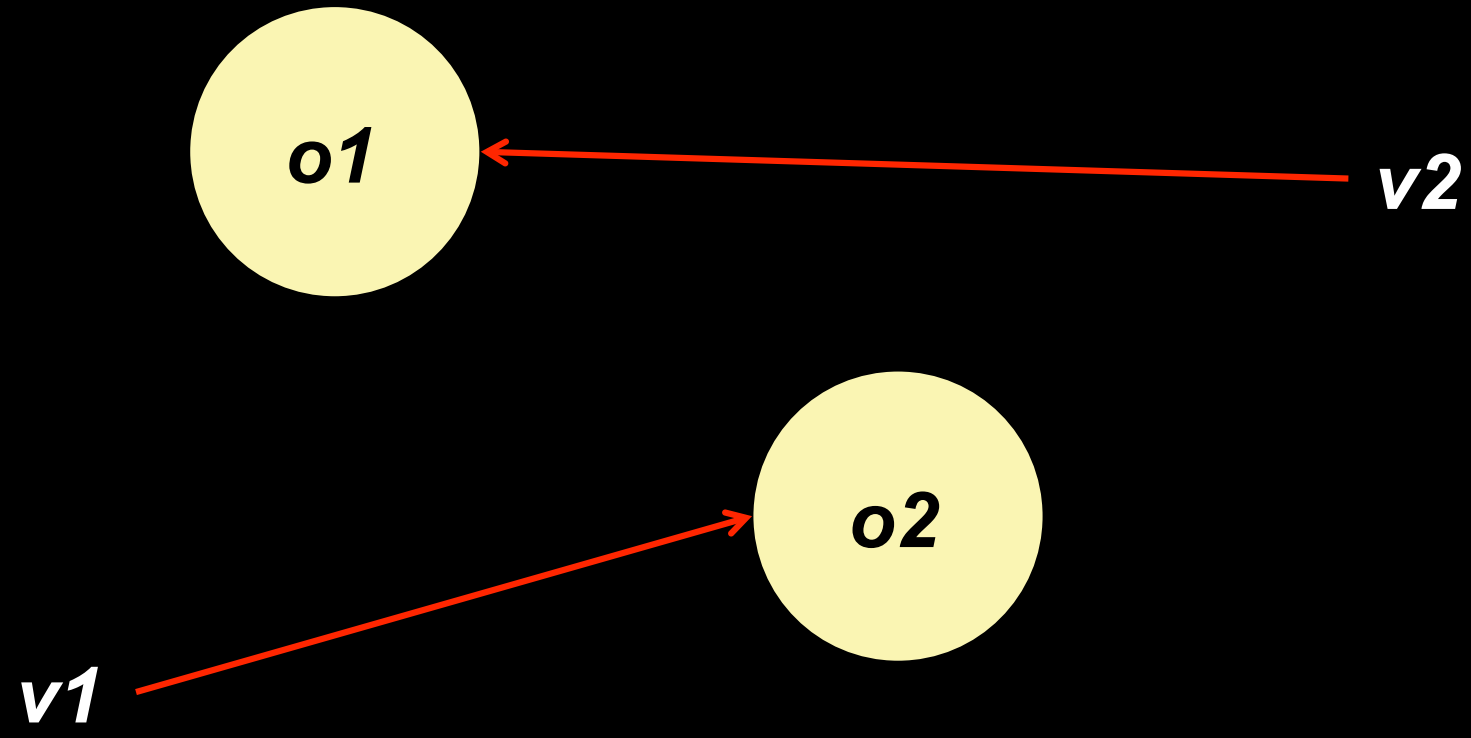
Echange d'identité !



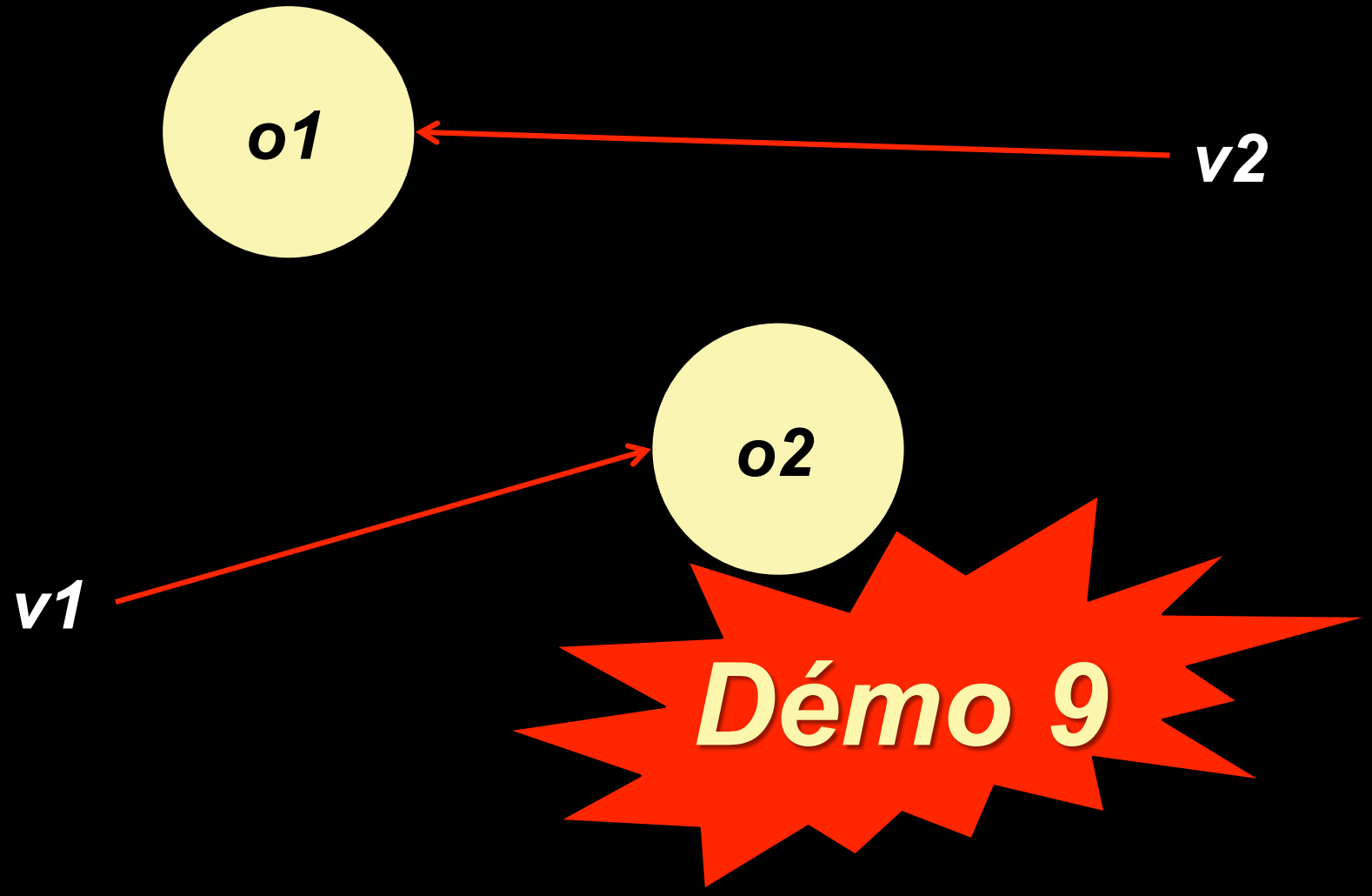
Echange d'identité !



Echange d'identité !



Echange d'identité !



Pour en savoir plus

Programmation Visuelle

- **Physical-eToys**

http://tecnodacta.com.ar/gira/projects/physical-etoys/

- **Phidgetlab**

http://www.hpi.uni-potsdam.de/hirschfeld/projects/phidgetlab/

- **SqueakBot**

http://wiki.laptop.org/go/Projects/SqueakBot

Smalltalks dédiés

- **NxTalk** : Lego Mindstorm

http://www.hpi.uni-potsdam.de/hirschfeld/projects/nxtalk/

- **Smalltalk pour PIC**

http://www.huv.com/uSeeker/smalltalk/pic.html

- FoxTalk

http://foxtalkbots.com/

Robotique @ Douai

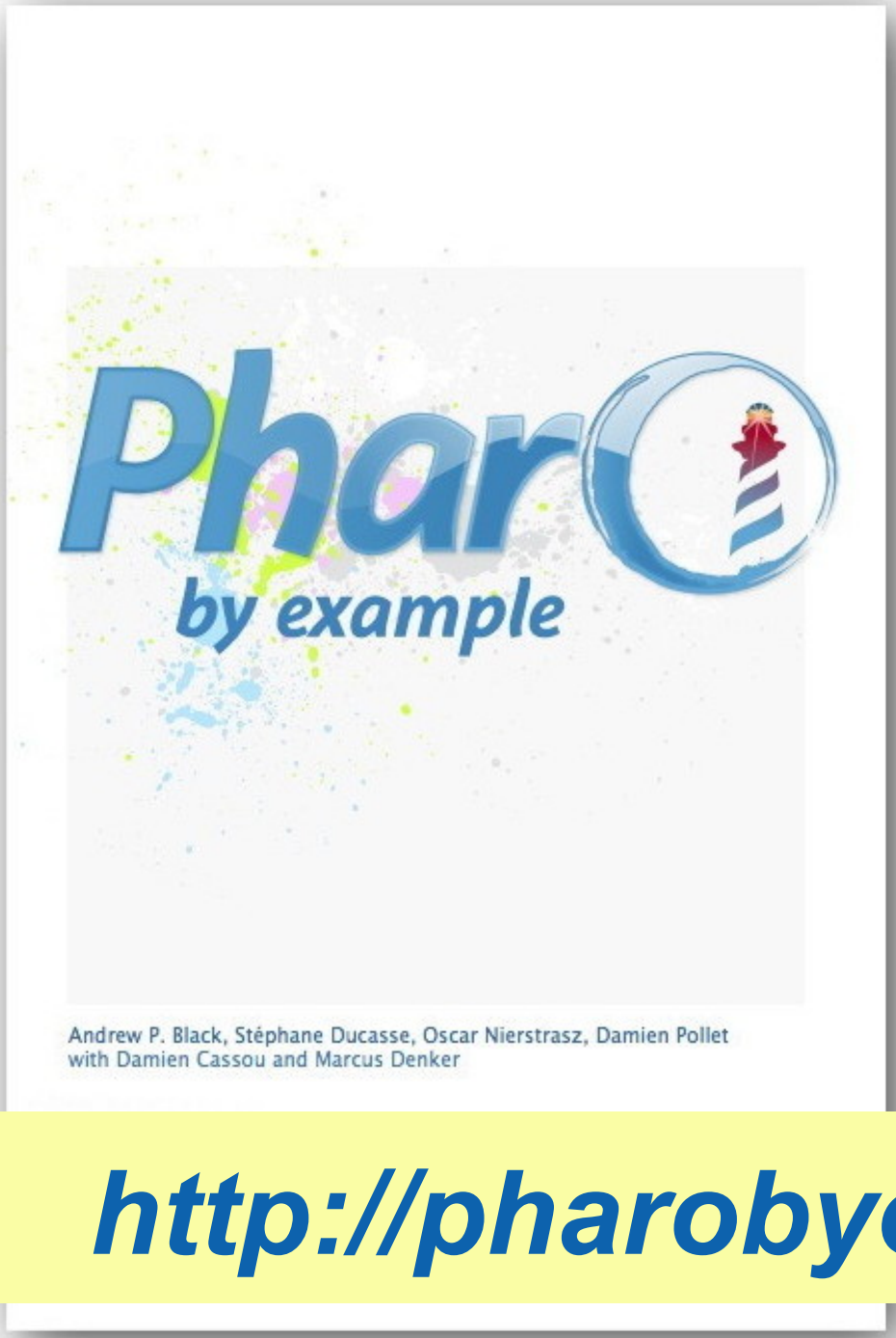
WifiBotST

http://vst.mines-douai.fr/WifiBotST

Articles, Code, Vidéos

http://vst.mines-douai.fr/Robotics

http://www.pharo-project.org/



LE Livre

http://pharobyexample.org/

Les VIDEOS

http://www.pharocasts.com/

A vos agenda !

Organisé par Linux Azur

Atelier sur 

Vendredi 17 juin 2011

Atelier animé par ***ramix***

René Mages

Robots Mobiles & Autonomes

avec

Pharo

Noury Bouraqadi
http://car.mines-douai.fr/noury

JM2L – 27 nov 2010