

Environnements de développement de logiciel

J.F. Santucci
UMR CNRS 6134 SPE
santucci@univ-corse.fr

EDI DrRacket : langage orienté objet

Ecosystème Racket

- Langage de base : Scheme
 - DrRacket : application Racket et EDI
 - Documentation : <http://docs.racket-lang.org>
 - Fichiers de départ :
`#lang racket`
`;;` Ecrire vos programmes.
-

Racket et les types

- Par défaut : Langage sans type
 - possibilité d'ajouter des vérifications sur les types
 - Documentation : <http://docs.racket-lang.org/ts-guide/index.html>
-

Racket et POO

Racket supporte la programmation objets avec les notions

- de classes,
 - d'héritages,
 - de constructeurs,
 - de surcharges de méthodes et
 - de portée des méthodes (publiques, privées)
-

P.O.O

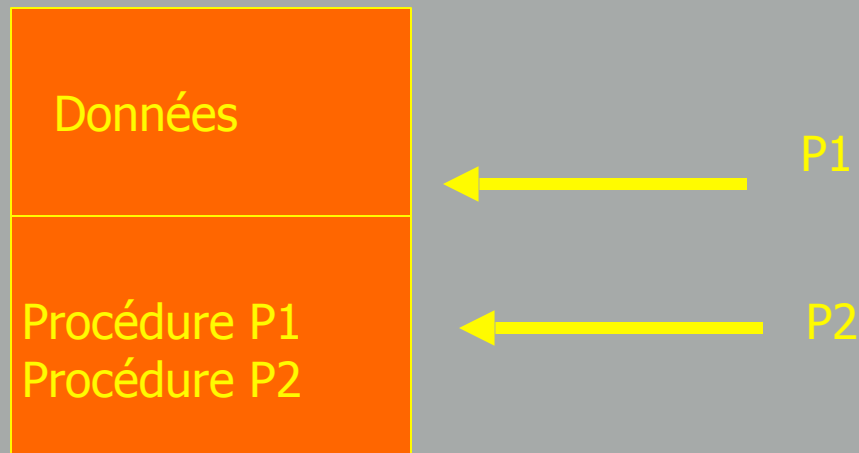
Notion d'objets

- Dans les deux cas, principes de base :
 - . Encapsulation
 - . Communication
 - . Création d'exemplaires
 - . héritage
- Élément de base : l'objet

P.O.O

Concepts de base

Une encapsulation est un regroupement sous un même nom de données et des procédures qui les manipulent.



P.O.O

Concepts de base

- Communication
 - . Par envoi de messages : on active les points d'entrée des encapsulations en leur envoyant des messages.
 - . A la réception d'un message, l'encapsulation exécute la procédure correspondante.
- Création d'exemplaires
 - . Par instanciation : classes
- Héritage : héritage de propriétés entre classes.

P.O.O

Encapsulation

- Classiquement en informatique, on distingue deux entités bien distinctes : le code et les données.
- On déclare les données sans se soucier de l'utilisation qui en sera faite.
- Chaque donnée est donc visible de toutes les procédures.

P.O.O

Encapsulation

- Grâce à la notion d 'encapsulation, une procédure ne peut manipuler que les données appartenant à une encapsulation
- Elle ne peut pas modifier les données d 'une autre encapsulation.

P.O.O

Encapsulation

- Vue de l'extérieur une encapsulation est :
 - . Une région mémoire
 - . Une entité unique et indépendante
 - . Une liste de points d'entrée.
- Vue de l'intérieur une encapsulation est :
 - . Des données structurées locales
 - . Des procédures locales correspondant chacune à un point d'entrée et qui manipulent ces données.

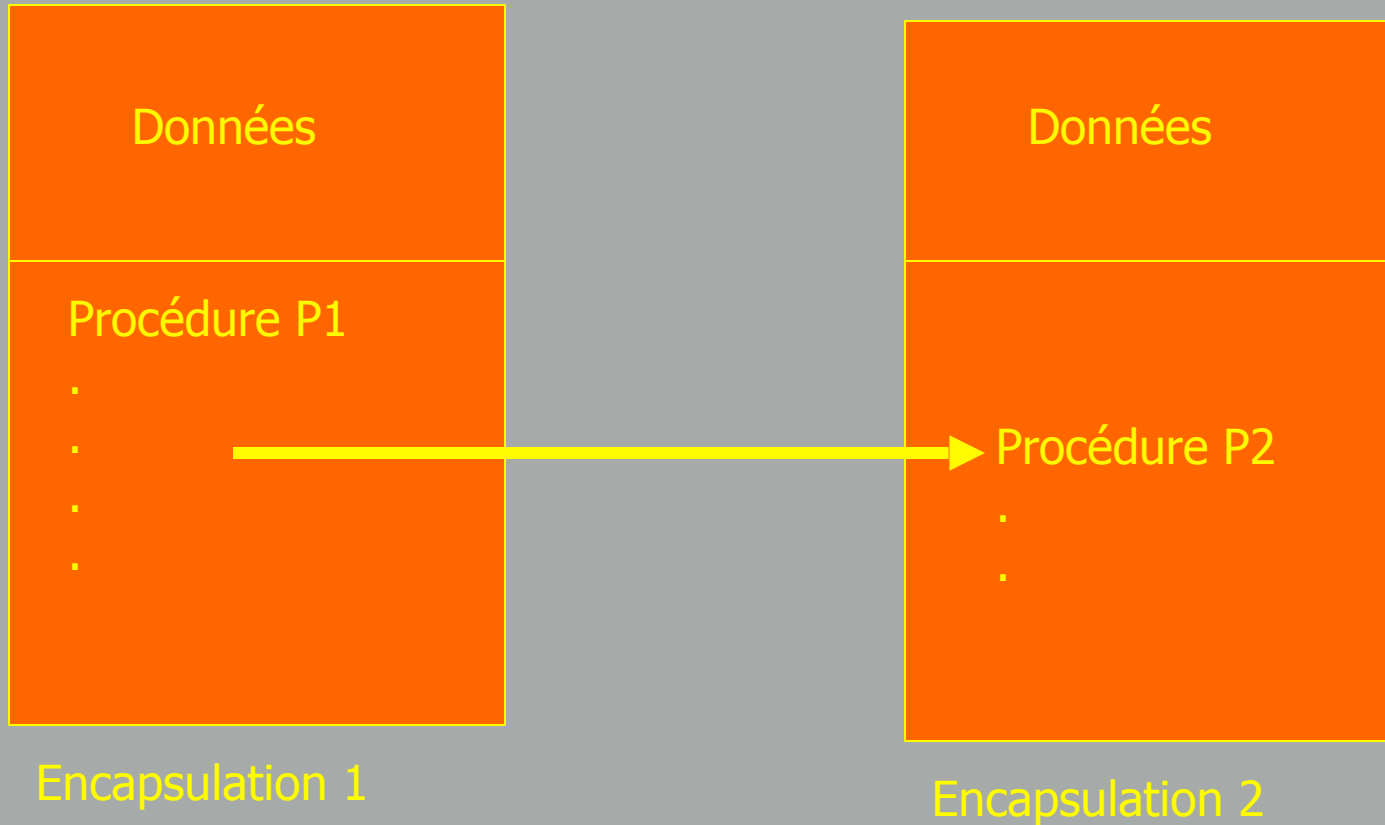
P.O.O

Envoi des messages

- Une transmission de message doit spécifier :
 - . Le nom de l'encapsulation destinataire
 - . Le nom du point d'entrée à activer
 - . Des arguments éventuels si deux encapsulations s'échangent des valeurs.
- Le message ne sait rien de l'implémentation du destinataire. Il ne connaît que son nom.
- Cela permet de modifier la représentation interne sans changer les programmes appelants.

P.O.O

Envoi des messages



P.O.O

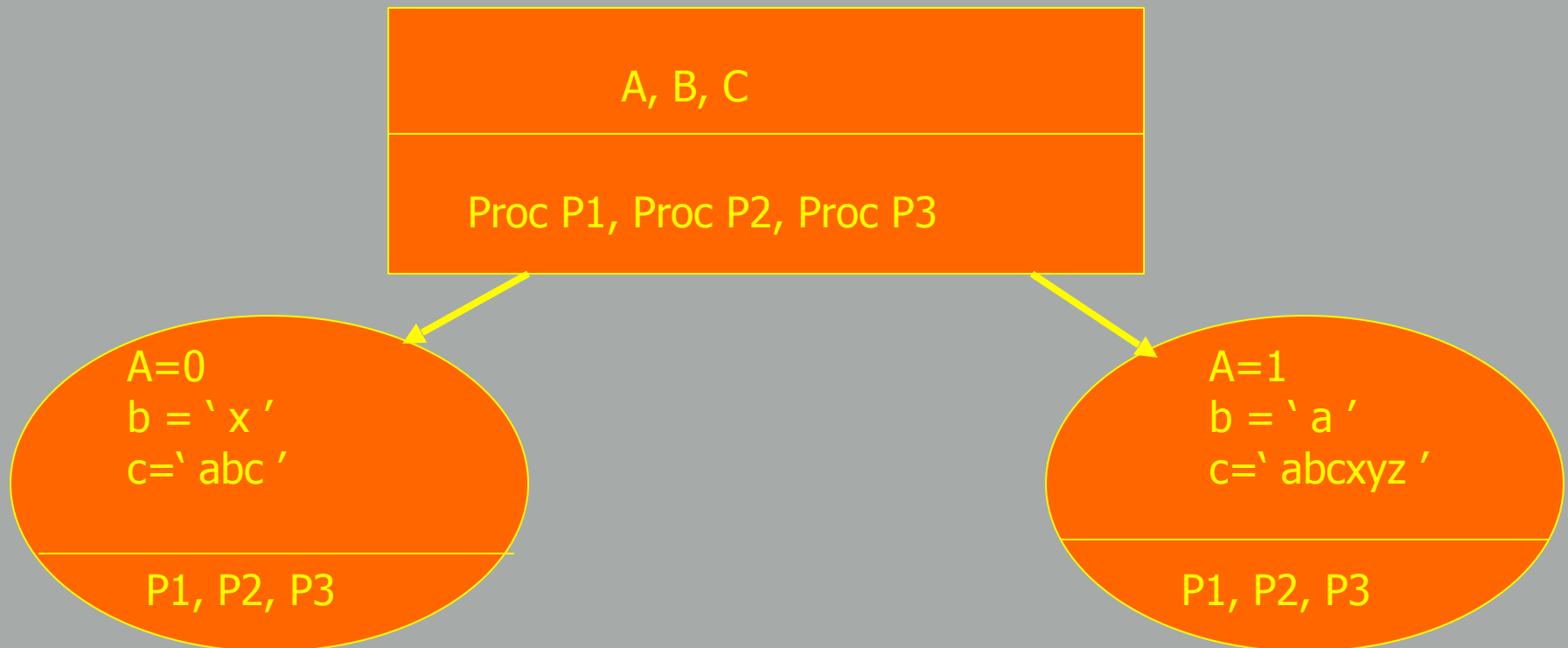
Création d'exemplaires

- Les objets ayant un même comportement seront représentés par un prototype qui définira leurs propriétés.
- Le prototype est une sorte de moule à partir duquel on fabrique autant d'éléments que l'on veut.
- Le prototype est un moyen de décrire un ensemble par définition.
- Les exemplaires d'un prototype sont les éléments de cet ensemble.
- Dans la terminologie des langages objets on parlera de classe.

P.O.O

Création d'exemplaires

Les exemplaires ainsi créés sont appelés instances de la classe à partir de laquelle ils sont fabriqués.



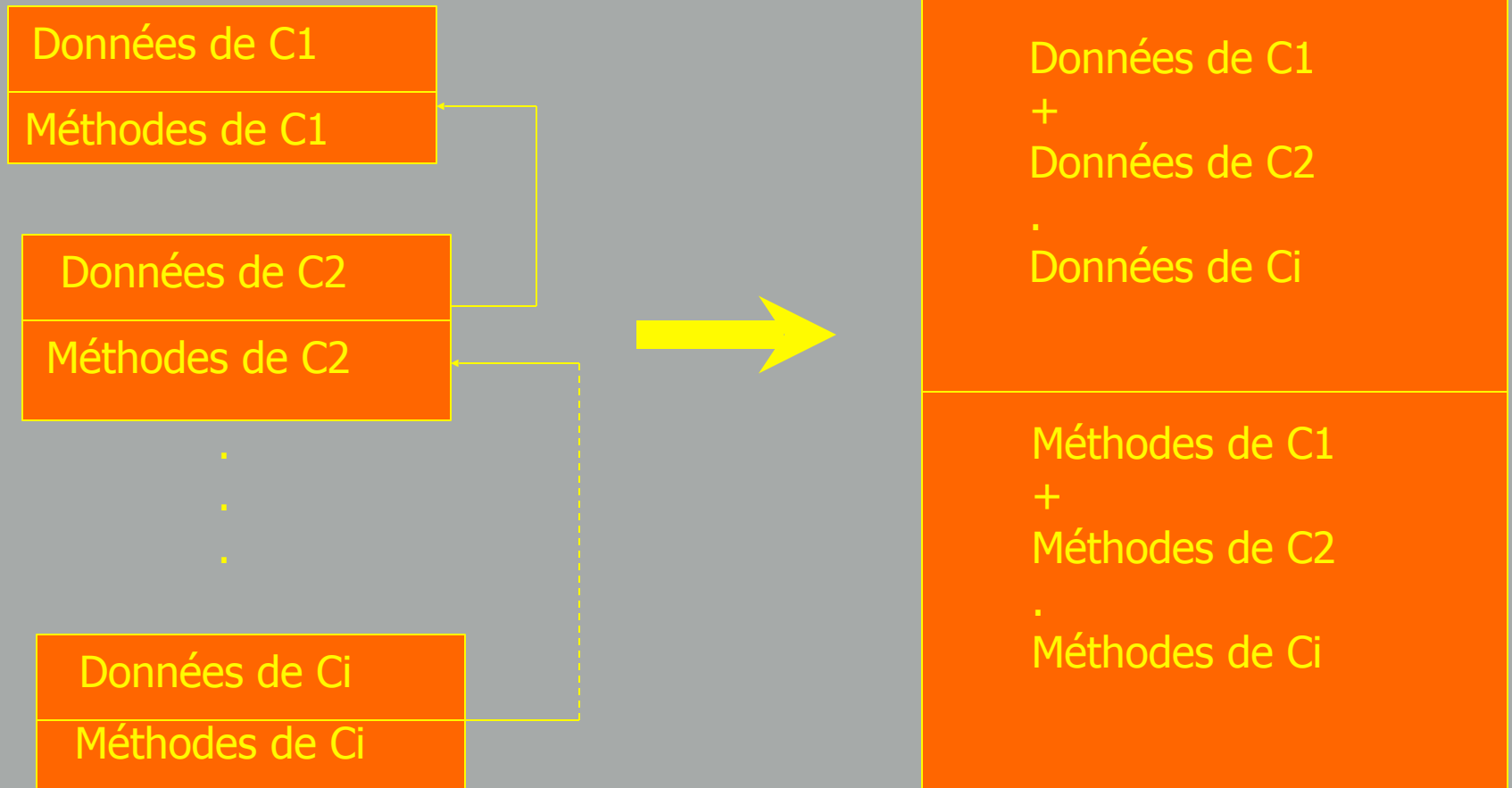
P.O.O

Héritage

- On peut avoir envie de définir une classe possédant toutes les propriétés d'une autre classe déjà définie, plus un certain nb d'autres.
- Le mécanisme d'héritage permet de ne définir que les propriétés particulières au nouvel objet.
- Créer un lien hiérarchique entre deux objets permet donc de faire hériter l'ensemble des données et des méthodes.

P.O.O

Héritage



P.O.O

Arbre d'héritage

- En conception orientée objet, tout objet du système que l'on crée doit être rattaché à un autre objet.
- Il existe dans un environnement de P.O.O. une classe prédéfinie : OBJET.
- Cette classe définit les fonctionnalités que doivent posséder toutes les classes.

P.O.O

Arbre d'héritage

- En fait tout environnement de P.O.O est constitué d'un arbre d'héritage comprenant un certain nombre de classes prédéfinies.
- Toutes ces classes héritent de la classe OBJET.
- Quand on veut fabriquer une nouvelle classe, celle-ci doit s'insérer dans cette pyramide.

P.O.O

Arbre d'héritage

- Toute nouvelle classe doit donc être fabriquée à partir d'une classe existante en créant un lien d'héritage.
- Une classe en P.O.O. ne peut en général avoir de lien d'héritage direct qu'avec un seul père : héritage simple.

P.O.O

Classe : exemple

Les données internes ne sont accessibles que par les procédures locales à une classe qui s'appelle des méthodes.

