

Etude de cas « Prison de Nantes »

Franck Barbier

Université de Pau et des Pays de l'Adour

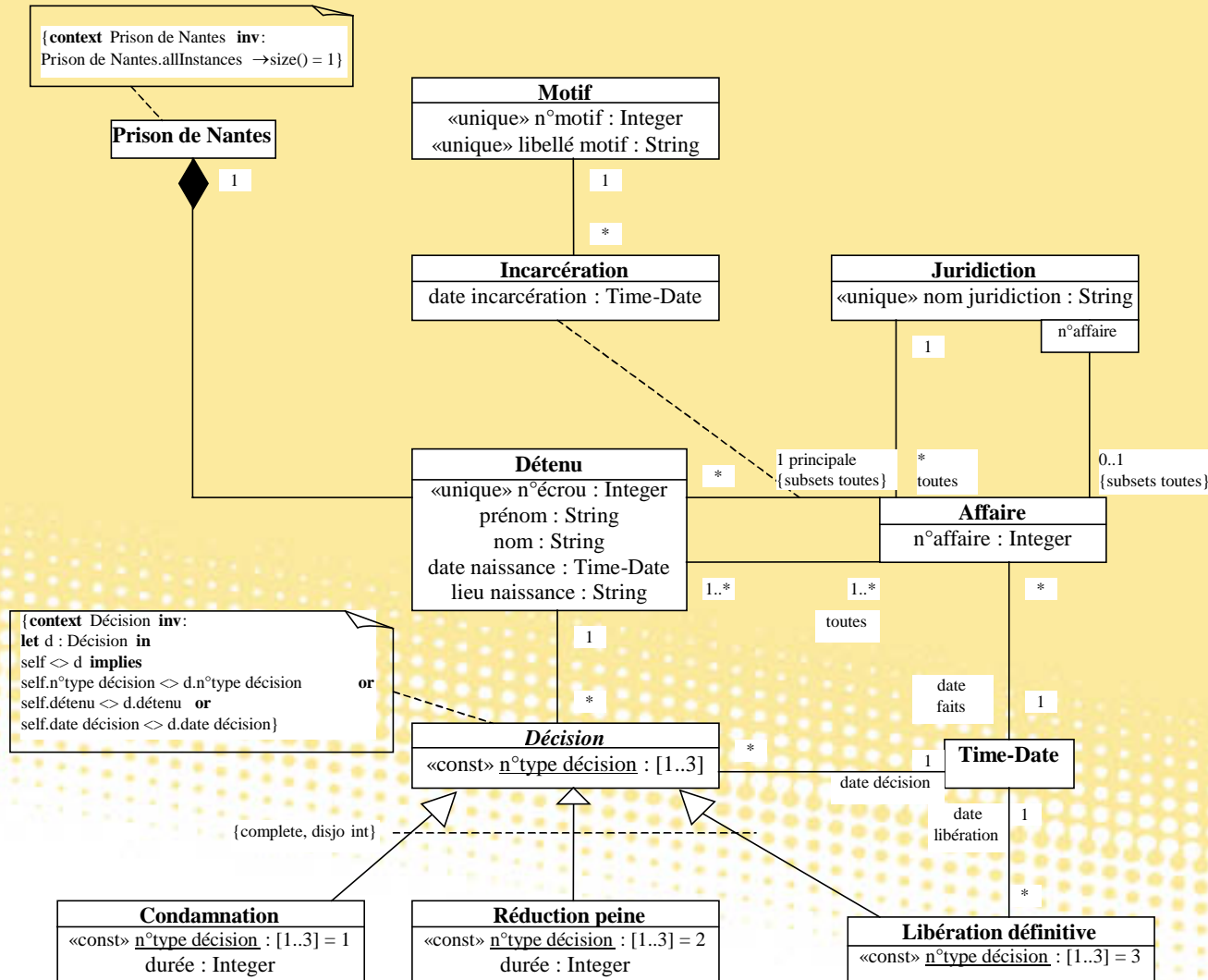
Téléchargement : www.PauWare.com

Tout commence par le langage naturel...

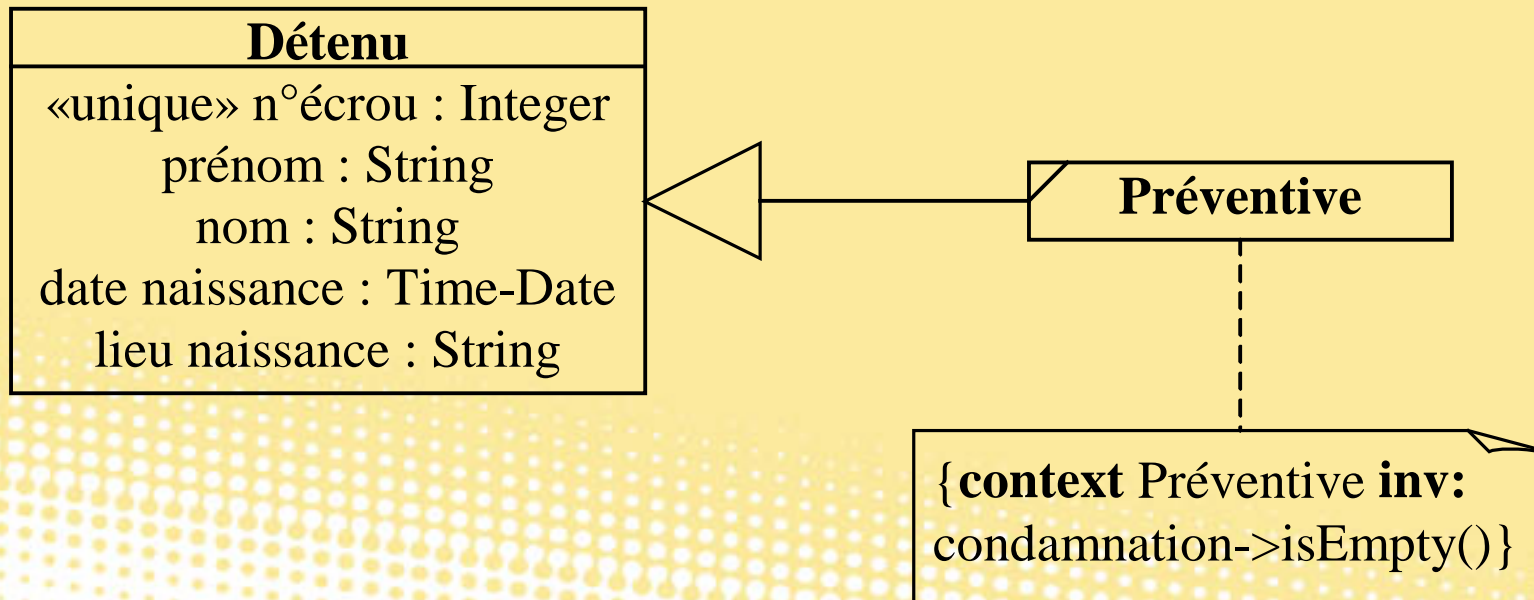
L'informaticien : « est-ce qu'un prisonnier peut arriver à la prison pour plusieurs affaires ? »

Le directeur de la prison de Nantes : « il peut être écroué en « préventive ». Dans ce cas, il n'a pas encore été condamné. Mais il subit une décision d'incarcération qui a trait à une affaire. Il a pu être écroué parce qu'il a été condamné pour une affaire. Il peut aussi être écroué parce qu'il a été condamné pour plusieurs affaires. »

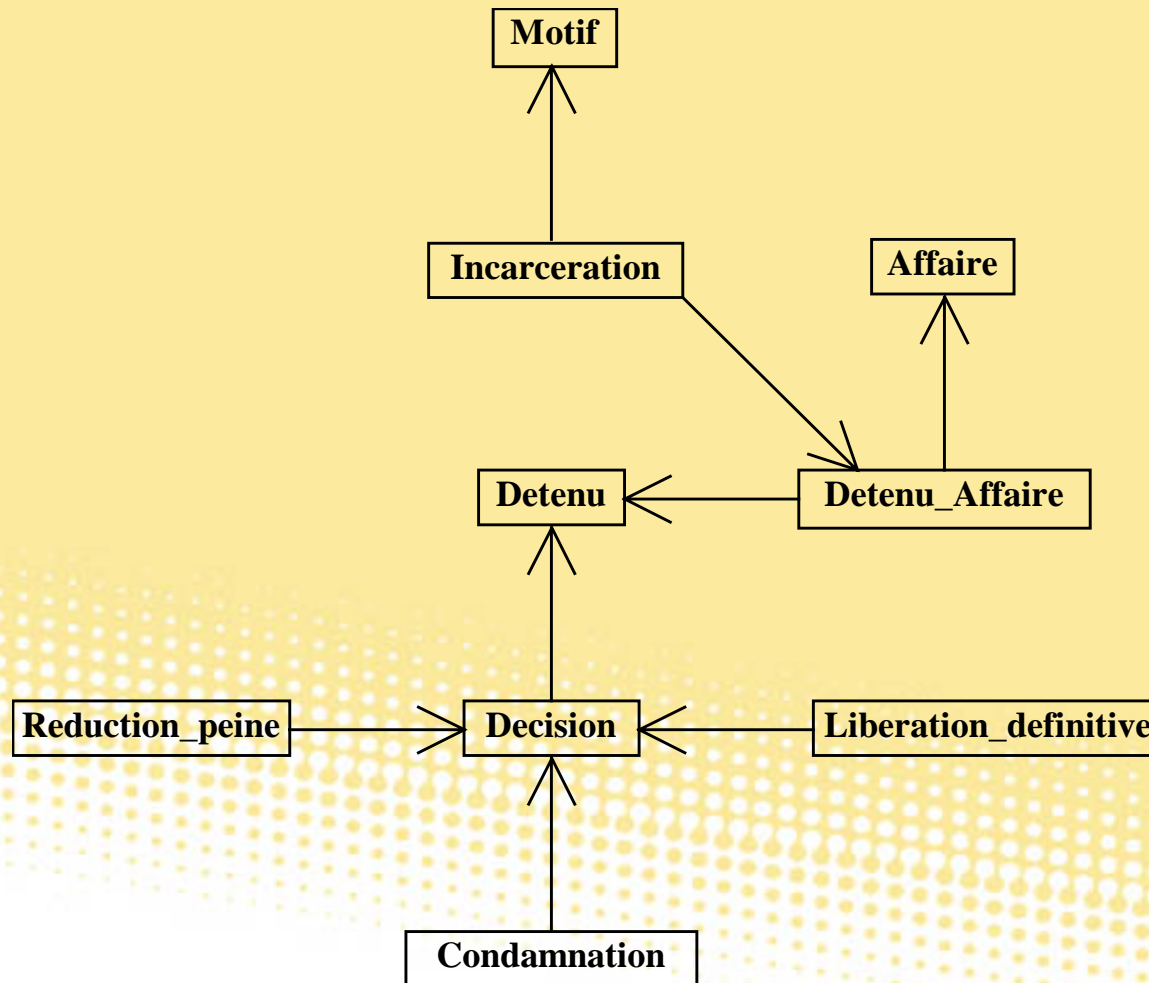
Prison de Nantes



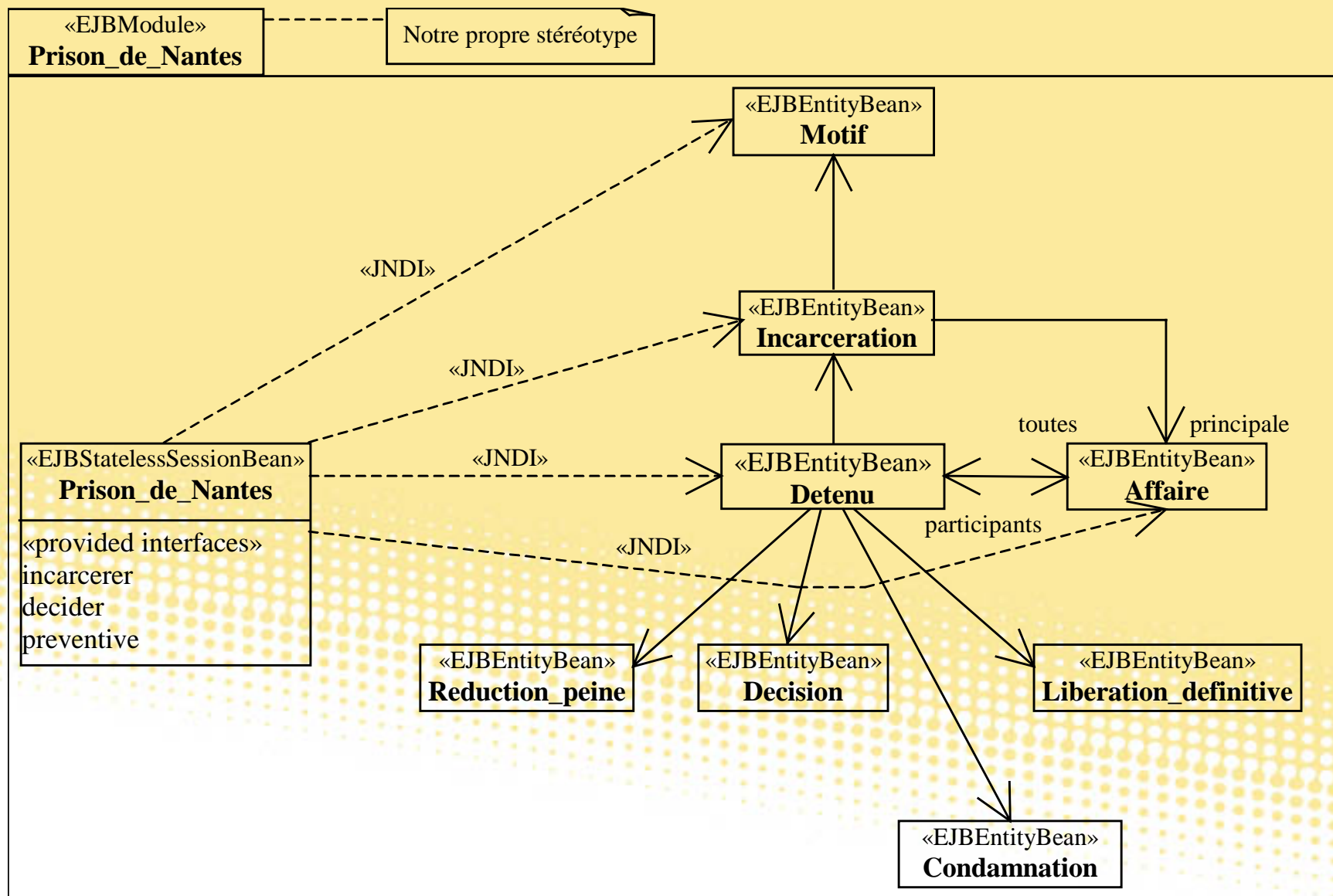
PIMs, rationalité des modèles



PSMs, tiers « persistance »



PSMs, tiers « intergiciel »



PSMs, différents niveaux

- Niveau « compatibilité *EJB Specification Version 2.1* »
- Niveau « serveur J2EE »
- Autres ?
- Comment marier (« tisser ») les modèles en conservant l'inter-cohérence ? NetBeans 5.0 n'y arrive pas...

PSMs niveau « *EJB Specification Version 2.1* »

Exemple *Detenu-Condammnation*

```
<ejb-relation>
  <description/>
  <ejb-relation-name>Detenu-Condammnation</ejb-relation-name>
  <ejb-relationship-role>
    <ejb-relationship-role-name>detenu</ejb-relationship-role-name>
    <multiplicity>One</multiplicity>
    <relationship-role-source>
      <ejb-name>DetenuBean</ejb-name>
    </relationship-role-source>
    <cmr-field>
      <cmr-field-name>condammnation</cmr-field-name>
      <cmr-field-type>java.util.Collection</cmr-field-type>
    </cmr-field>
  </ejb-relationship-role>
  <ejb-relationship-role>
    <ejb-relationship-role-name>condammnation</ejb-relationship-role-name>
    <multiplicity>Many</multiplicity>
    <relationship-role-source>
      <ejb-name>CondammnationBean</ejb-name>
    </relationship-role-source>
  </ejb-relationship-role>
</ejb-relation>
```

PSMs niveau « serveur J2EE » : *sun-cmp-mappings*

Exemple *Detenu-Condammnation*

```
<entity-mapping>
  <ejb-name>DetenuBean</ejb-name>
  <table-name>DETENU</table-name>
  ...
  <cmr-field-mapping>
    <cmr-field-name>condammnation</cmr-field-name>
    <column-pair>
      <column-name>DETENU.N_ECROU</column-name>
      <column-name>CONDAMNATION.N_ECROU</column-name>
    </column-pair>
    <fetches-with>
      <none/>
    </fetches-with>
  </cmr-field-mapping>
</entity-mapping>
```

Navigations, d'OCL à EJB QL

Exemple de la notion de « préventive »

- *SELECT DISTINCT OBJECT(d1) FROM Detenu d1, IN (d1.decision) as d2 WHERE d2.nTypeDecision = '1'*
- *SELECT DISTINCT OBJECT(d) FROM Detenu d WHERE d.condamnation IS NOT EMPTY*

Satisfaction des contraintes

■ Cardinalités

Opération d'interface « incarcérer » : l'incarcération a un et un seul motif et une et une seule affaire dite « principale »

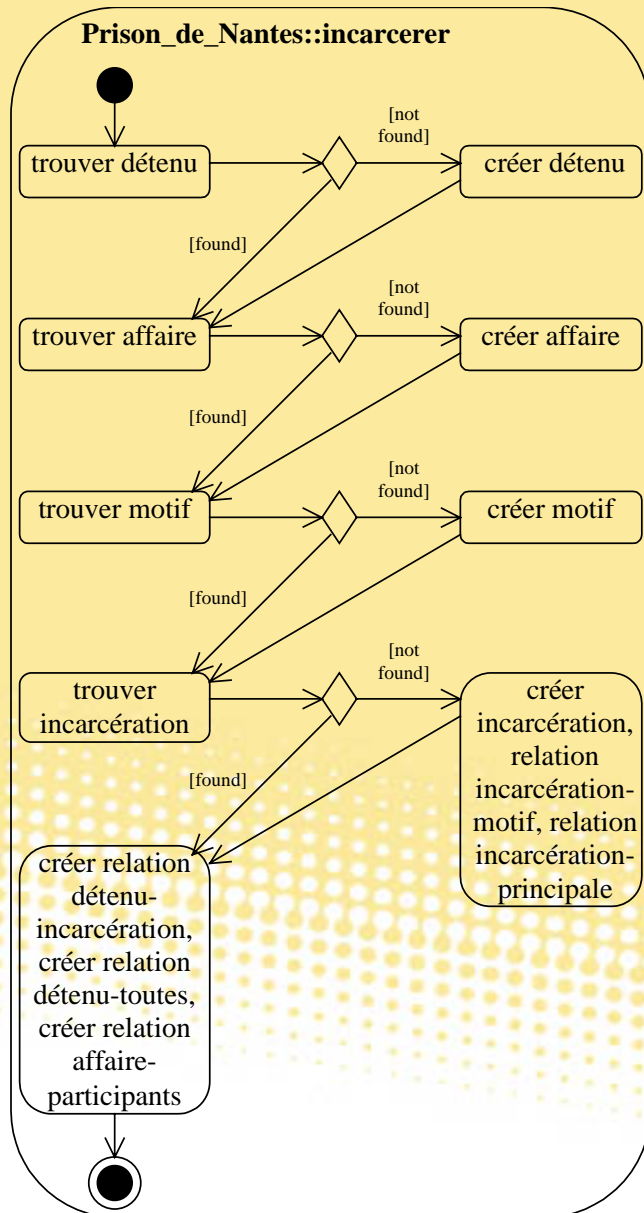
```
public interface IncarcerationLocalHome extends EJBLocalHome {
    IncarcerationLocal create(String nErou,java.sql.Timestamp dateIncarceration,MotifLocal motif,AffaireLocal principale) throws CreateException;
    ...
}
```

■ OCL

context *Detenu inv i1:*

toutes->includesAll(incarceration.principale)

```
public boolean i1() {
    return getToutes().contains(getIncarceration().getPrincipale());
}
```



Why not?

Les *Activity Diagram* ne sont jamais que des organigrammes. Ils peuvent éventuellement servir à décrire le contenu d'opérations d'interfaces des EJBs, ici la *business* méthode *incarcérer* du *Stateless Session Bean* *Prison_de_Nantes*

Exercice

- Implanter les relations EJB *Detenu-Reduction_peine* et *Detenu-Liberation_definitive*
- Implanter ensuite la *business* méthode « décider » de la *provided interface* de l'EJB *Prison_de_Nantes*, méthode qui crée une nouvelle décision et la rattache au détenu concerné