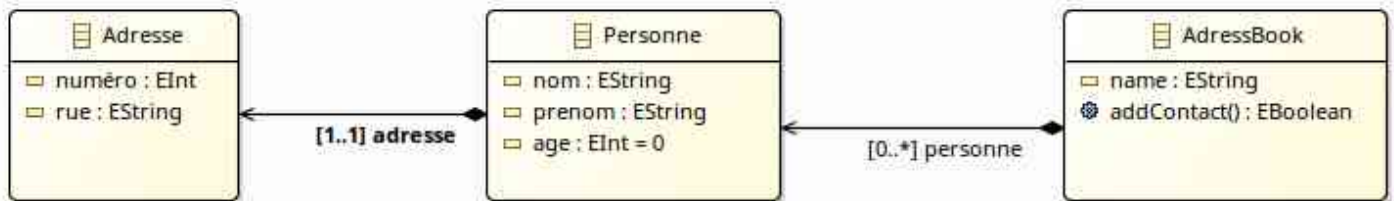
 S. Hérauvillle	Développement Collaboratif - UML	M1 GIL
	Création d'un projet ECORE	TP n° 1
		page 1 / 6

I - Présentation du sujet

Objectif : Développer une application Java en générant le code à partir d'un modèle UML.

Vous allez devoir développer une application permettant de gérer un carnet d'adresse (rudimentaire) défini par le schéma suivant :



Remarque : Le contenu de ce TP sera réutilisé dans le TP suivant.

II - Création du projet

Remarques :

- Vous trouverez en annexe une "liste d'astuces" à consulter avant de poser une question !
- Conseil : Lorsqu'Eclipse exécute une commande, il est préférable de le laisser terminer ses opérations avant de déclencher une autre opération.
- Pour réaliser le TP, vous aurez besoin des plugins "Modeling" d'Eclipse (Acceleo / EMF / Ecore / OCL), qui à priori sont déjà disponible sur votre environnement
- Utilisez la perspective : Modeling

- Créez un nouveau projet : Eclipse modeling framework → Ecore Modeling Project


nommez le projet : uml.addressbook
main package name : addressbook
namespace URI : addressbook.univrouen.fr
namespace Prefix : addressbook
Viewpoints : Design

- Sélectionnez dans l'arborescence, le fichier "addressbook.ecore"
clic droit → Initialize Ecore Diagram → Design → Entities
Sélectionnez le package "addressbook"

- Dessinez le schéma en utilisant la palette de droite pour obtenir les éléments souhaités
Le paramétrage des éléments peut s'effectuer par la fenêtre propriétés affichée en dessous,
ou par une fenêtre d'édition que l'on ouvre par un double-clic sur l'élément souhaité.

Remarques :

- Les attributs et méthodes sont disponibles dans l'onglet "Feature" de la palette,
- Pour les attributs, utilisez les formats EString, EInt, ...
- Pour afficher la fenêtre graphique lorsque l'on revient sur le projet, il suffit de faire un double-clic sur le fichier entité "addressbook", disponible lorsque l'on développe addressbook.aird → Design

 UNIVERSITÉ DE ROUEN <i>S. Hérauvill</i>	Développement Collaboratif - UML	M1 GIL
	Création d'un projet ECORE	TP n° 1
		page 2 / 6

- Lorsque le schéma est réalisé, sauvegardez votre projet.
- Sélectionnez le fichier "addressbook.ecore", et affichez le dans la fenêtre centrale au moyen d'un double-clic. Développez et analysez son contenu.
- Sélectionnez le fichier "addressbook.ecore" dans l'arborescence de la fenêtre de gauche
A l'aide du menu contextuel, ouvrez le projet avec les outils suivants :
 - Open With
 - OCLinEcore Editor
 - XML Editor

Quelles sont les informations affichées par ces outils, et sous quel format ?

III - Génération du code

- Sélectionnez le modèle "addressbook.genmodel" dans l'arborescence de gauche.
Remarque : si le genmodel n'est pas déjà présent, il est possible de le générer à partir du modèle ecore.

Ouvrez le genmodel dans la fenêtre centrale (double-clic) et sélectionnez le package "Addressbook"

Dans la fenêtre de propriétés (en bas d'écran) : All → Base Package

Définissez le package de base : addressbook.univrouen.fr

- Sélectionnez le package "Addressbook"
clic-droit → Generate Model Code
L'ensemble du projet est généré en Java
- Proposez une analyse des contenus des packages créés, et de leur organisation.

Nous allons maintenant modifier le schéma UML.

Question III.1 - Ajout d'un attribut dérivé

Ajoutez à la classe "Personne", un nouvel attribut nommé "identifier" de type String.

Paramétrez les propriétés avec les valeurs suivantes :

```

derived = true
changeable = false
transient = true
volatile = false

```


Justifiez ce paramétrage.

Question III.2 - Ajout d'une méthode

Ajoutez également une méthode "display" à la classe Personne. Cette méthode n'attend pas d'argument, et retourne une chaîne de caractères.

Question III.3 - Actualisation du projet

Enregistrez les modifications effectuées précédemment.

 <i>S. Hérauville</i>	Développement Collaboratif - UML	M1 GIL
	Création d'un projet ECORE	TP n° 1
		page 3 / 6

- Si lorsque vous visualisez le contenu du modèle (`addressbook.genmodel`), vous vous apercevez que les modifications ne sont pas encore prise en compte, il faut alors actualiser le projet.
- Faites une mise à jour du modèle (`addressbook.genmodel` → `reload`) et validez son contenu.
- Relancez ensuite la génération du code, et vérifiez également que le code a bien été réactualisé.

Question III.4 - Insertion de code

- En éditant directement le fichier `PersonneImpl.java`, modifiez le code comme suit :

* La méthode `getIdentifieur` doit retourner une chaîne de caractères formée par l'ensemble des attributs de la classe, sous la forme : `nnnpppAAA`

où `nnn` représente les 3 premières lettres du nom
 `ppp` représente les 3 premières lettres du prénom
 `aaa` représente l'age


* La méthode `display` doit retourner une chaîne de caractères formée comme suit :

Nom : `nom`
 Prénom : `prenom`
 Age : `age`
`nom, prenom et age` étant les attributs de la classe.

- Modifiez l'annotation devant ces 2 éléments en `"@generated NOT"`
- De la même manière, ajouter une méthode `display` à la classe `AdresseImpl`
- Enregistrez les modifications.

Question III.5 - Pérennité des modifications

- Fermez la fenêtre d'édition de la classe `PersonneImpl.java`
- Revenez au modèle UML et ajoutez une méthode `"removeContact"` à la classe `AddressBook`
- Relancez la génération du code.
- Vérifiez maintenant que ces dernières modifications sont bien effectuées et sont visible dans le code généré, mais aussi que le code saisi précédemment (§III.4) est toujours présent.

 <i>S. Hérauvill</i>	Développement Collaboratif - UML	M1 GIL
	Création d'un projet ECORE	TP n° 1
		page 4 / 6

IV - Implémentation d'une classe de test

Question IV.1 - Création d'un carnet d'adresse

- Le code généré propose une classe "Factory" disponible pour la mise en œuvre, que nous allons utiliser.

- Sélectionnez le projet `uml.addressbook`

Ajoutez dans les dépendances du projet, le package `org.junit`

- Ajoutez un package : `uml.addressbook.model.test`, contenant la classe `AddressbookTest`

- Créez une méthode de test avec le code suivant :

```
@Test
public void createAddressbookTest() {
    Addressbook carnet = AddressbookFactory.eINSTANCE.createAddressbook();
    carnet.setName("MonCarnet");

    // Création d'une adresse
    Adresse adr1 = AddressbookFactory.eINSTANCE.createAdresse();
    ...


    // Création d'une personne
    Personne pers1 = AddressbookFactory.eINSTANCE.createPersonne();
    ...

    // Ajout personne / adresse dans carnet
    ...

    // Affichage du contenu de carnet - affichez les valeurs suivantes :

    carnet.getNom()
    carnet.getPersonne().size()
    carnet.getPersonne().get(0).display()
    carnet.getPersonne().get(0).getAdress().display()
```

- Complétez le code afin d'insérer 3 personnes dans le carnet d'adresse, et implémentez les tests à réaliser pour valider le résultat obtenu.

 UNIVERSITÉ DE ROUEN <i>S. Hérauvill</i>	Développement Collaboratif - UML	M1 GIL
	Création d'un projet ECORE	TP n° 1
		page 5 / 6

Question IV.2 - Sauvegarde du carnet d'adresse

- La question précédente à permis de générer un carnet d'adresse avec quelques entrées. Il est utile de pouvoir sauvegarder le contenu, ce que vous allez réaliser maintenant, en utilisant un format XML.

- Analysez la portion de code fournie ci-dessous :

```
ResourceSet resourceSet = new ResourceSetImpl();

resourceSet.getResourceFactoryRegistry().getExtensionToFactoryMap().
    put("addressbook", new XMIResourceFactoryImpl());

Resource resource = resourceSet.createResource(uri);
resource.getContents().add(createAddressbook);
resource.save(null);
```


- Complétez le code afin de créer un fichier XML sauvegardant le contenu du carnet d'adresses.

Remarque : Le package `org.eclipse.emf.ecore.xmi.impl.XMIResourceFactoryImpl` est disponible sur universitice.

- Testez et validez le contenu du fichier

Question IV.3 - Restauration du carnet d'adresse

- Proposez un code permettant de restaurer le contenu du carnet d'adresse, à partir du fichier XML.

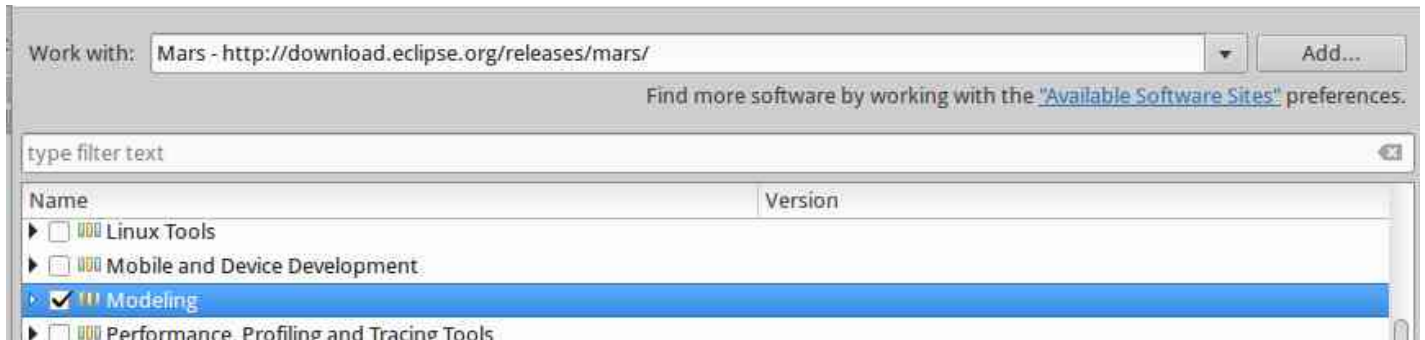
 S. Hérauvillle	Développement Collaboratif - UML	M1 GIL
	Création d'un projet ECORE	TP n° 1
		page 6 / 6

Annexes : Quelques astuces d'utilisation ...

- Installation des plug-ins

Remarque : Les plug-ins sont déjà installé sur votre poste étudiant.

help → new software → work with : Mars – <http://download.eclipse.org/download/releases/mars>
cocher le package "Modeling" → installation de 68 éléments !



- Génération d'un genmodel

Sélectionner le fichier "addressbook.ecore" dans l'arborescence

clic-droit → new → other → Eclipse Modeling Framework → EMF Generator Model

définir le nom du fichier géré addressbook.genmodel

Sélectionner le format du modèle de base Ecore model

Sélectionner le modèle ecore du projet

Sélectionner le package à convertir addressbook

- Modification du package

Sélectionner genmodel →	package	nom du package utilisé	addressbook
	base package	préfixe utilisé	univrouen

package défini dans les fichiers java => univrouen.addressbook

- Ajout d'une dépendance au projet

Dans l'arborescence du projet, sélectionner META-INF → MANIFEST.MF

onglet dépendance

Add → Sélectionner la dépendance à ajouter

- Affichage du schéma UML

Pour retrouver l'affichage du schéma UML lorsque la fenêtre a été fermée :

Sélectionner (double-clic) dans l'arborescence (sur) le fichier

addressbook.aird → design → entities → addressbook