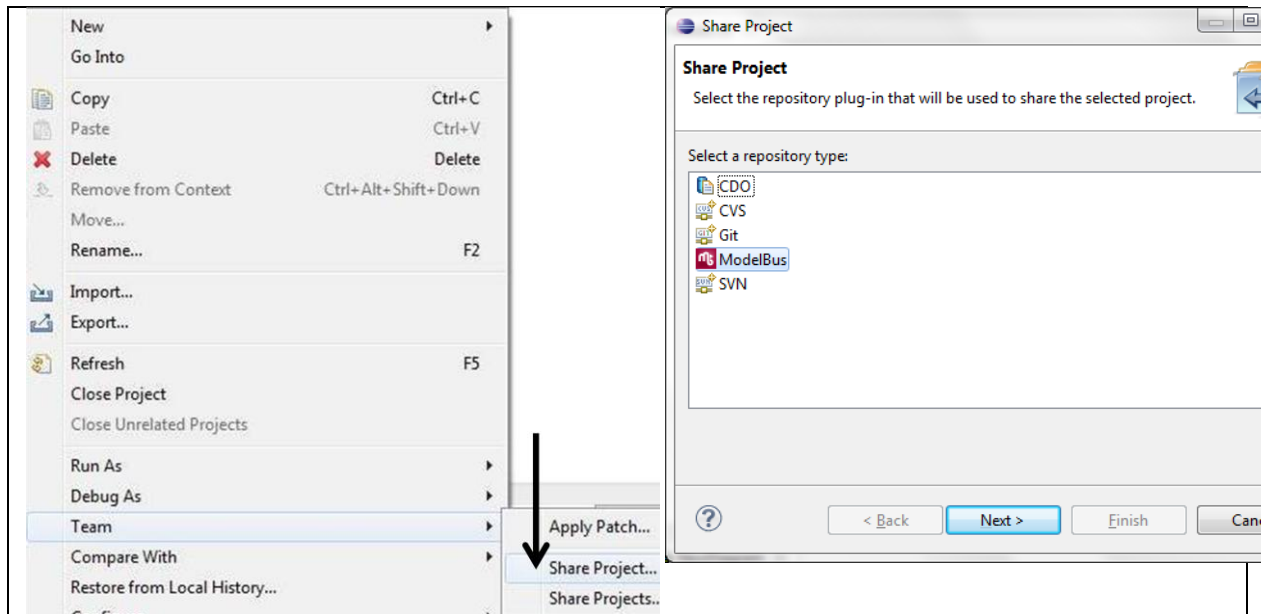
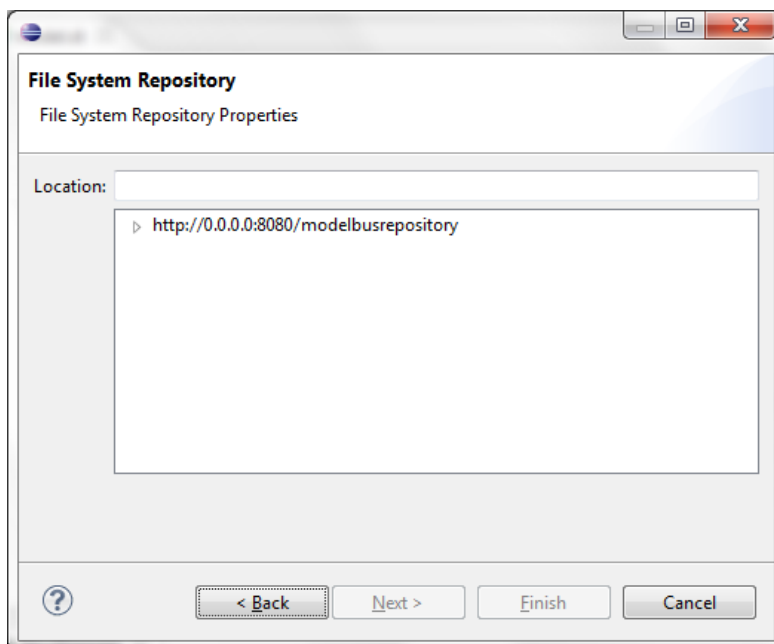


Partie 2 : utilisation d'un repository local

On clique droit sur notre projet essai dans la fenêtre Project Explorer pour accéder au menu Share Project..., puis on sélectionne ModelBus.

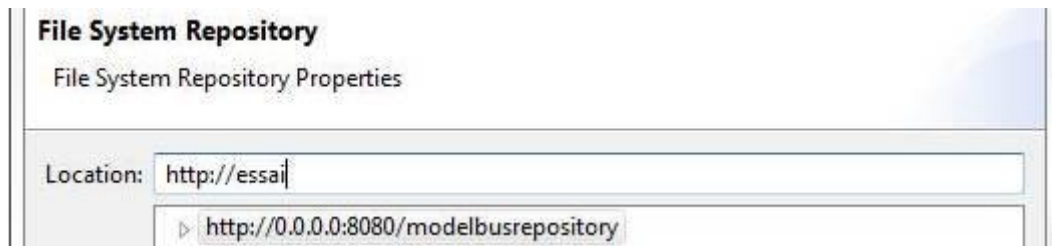


On arrive à la fenêtre suivante :

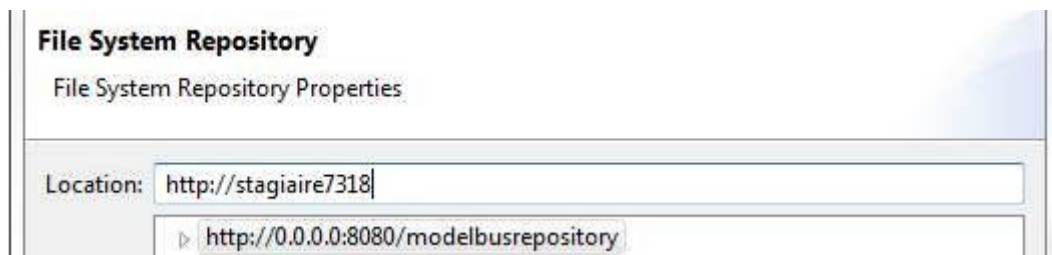


Attention : à ce niveau, nous pouvons vérifier si notre configuration fonctionne. Si on peut sélectionner la ligne `http...modelbusrepository`, c'est que tout va bien. Sinon, il faut retourner à la partie portant sur les préférences modelBus.

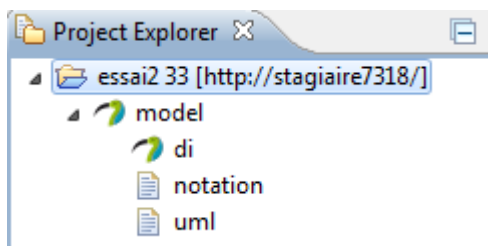
En sélectionnant la ligne en http, on obtient :



Si nous validons, on utilisera dans le repository, le même nom que pour notre projet qui est en local : `essai`. Pour aider la suite, nous allons changer <http://essai> par <http://stagiaireX>, en remplaçant x par le numéro qui vous a été attribué.



Une fois validé nous obtenons dans Project Explorer :

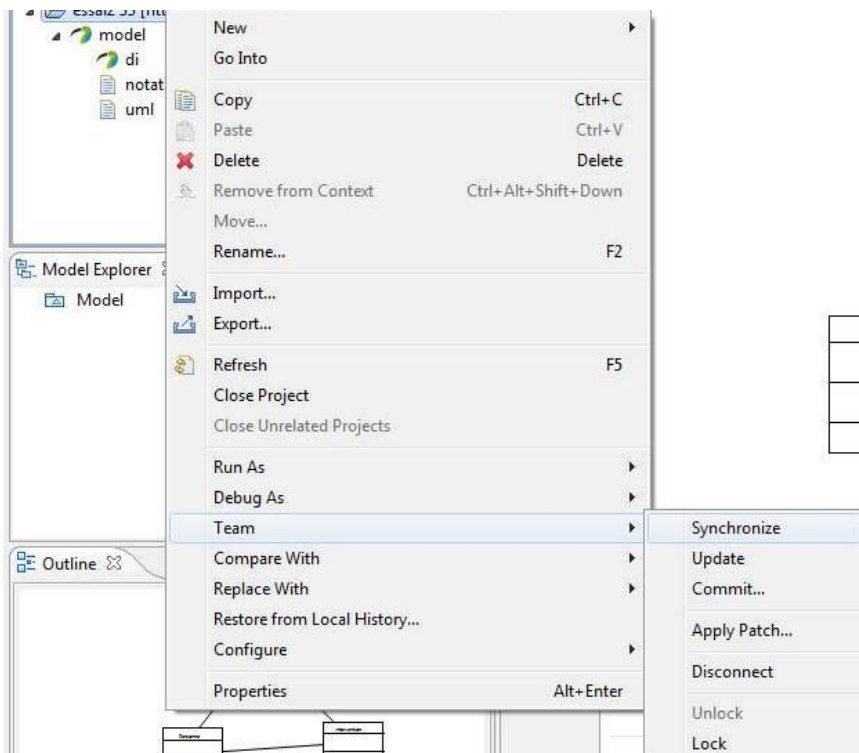


(sur le graphique vous avez `essai` à la place de `essai2 ...` parce que j'ai du recommencer, mais tout va bien).

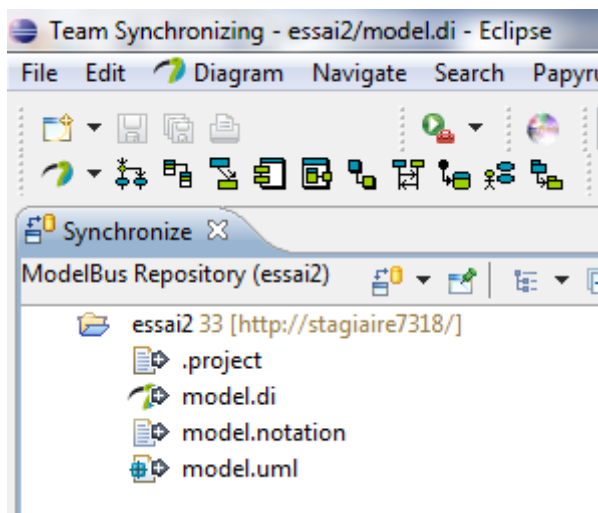
On observe le lien entre le nom du projet en local et le nom du projet dans le repository. L'erreur serait de penser que du côté repository nous avons la même chose que du côté local. Pour l'instant nous avons juste un lien entre les deux projets, mais du côté repository c'est vide.

Nous allons maintenant copier notre local sur le repository. La première chose à faire c'est de ... vérifier que du côté local, notre projet est bien enregistré, la copie que nous allons faire prendra notre enregistrement local et non ce qui est en cours dans Eclipse ! Donc on sauvegarde.

On va maintenant synchroniser notre projet local avec le repository. On repasse par la même manipulation que tout à l'heure (clique droit sur le projet), puis team, mais au lieu de share project... on va sur synchronize :

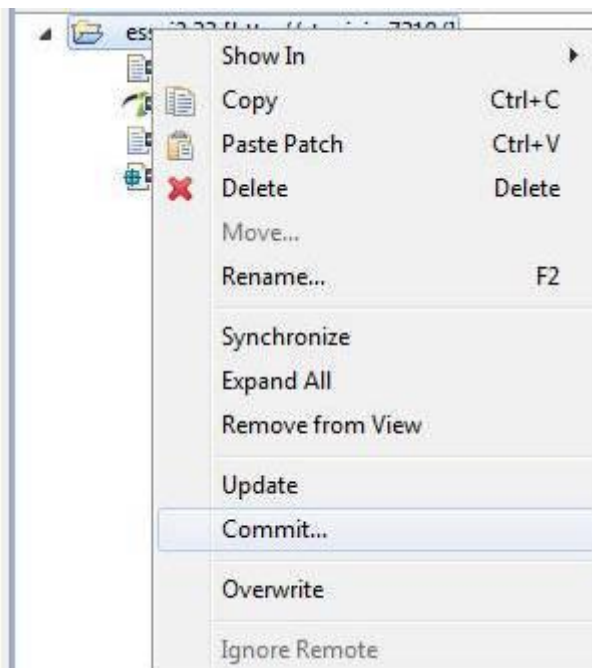


On accepte la fenêtre qui apparaît ensuite pour changer de perspective en basculant sur team synchronizing. La fenêtre project explorer est remplacée par la suivante :

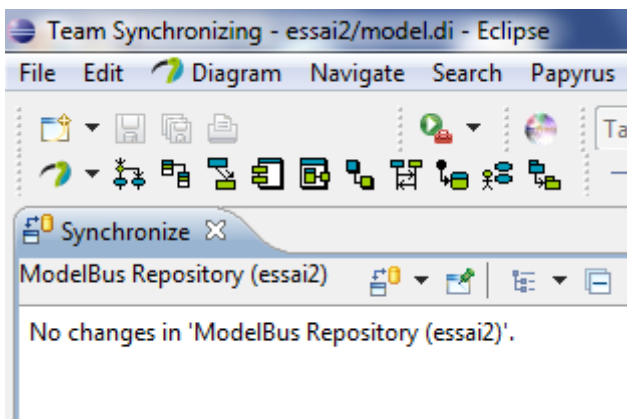


C'est ici qu'on peut observer si notre projet local, est différent ou égale au repository, et s'il y a des différences, dans quels sens sont-elles.

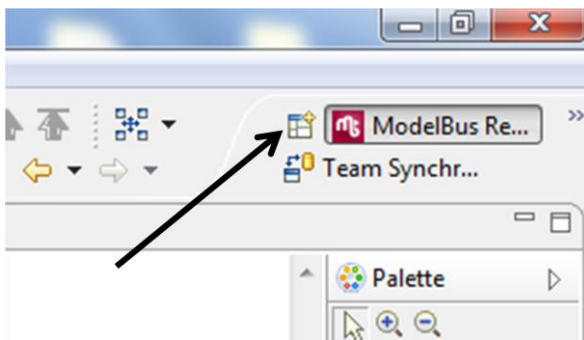
Dans notre cas, le projet local a des éléments qui ne sont pas dans le repository : les flèches en gris vont d'essai vers <http://stagiaire7318>. Pour mettre à jour notre repository (pour qu'il corresponde au projet local), on clique droit sur essai et on sélectionne commit :



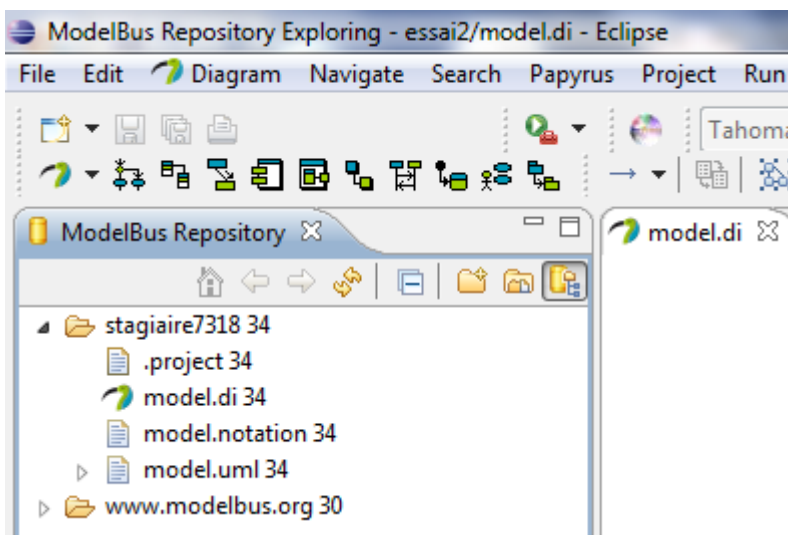
On peut ensuite mettre un commentaire dans la fenêtre qui apparaîtra. Nous arrivons ensuite à cet écran :



A ce stade, le projet local et le repository sont identiques. Changeons de perspective, pour aller explorer notre repository, en cliquant sur l'icône ci-dessous, et en sélectionnant ModelBus repository.



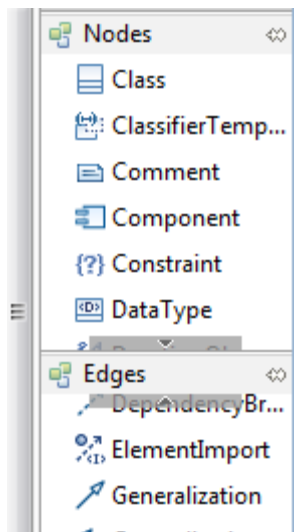
Nous arrivons à un écran de ce type :



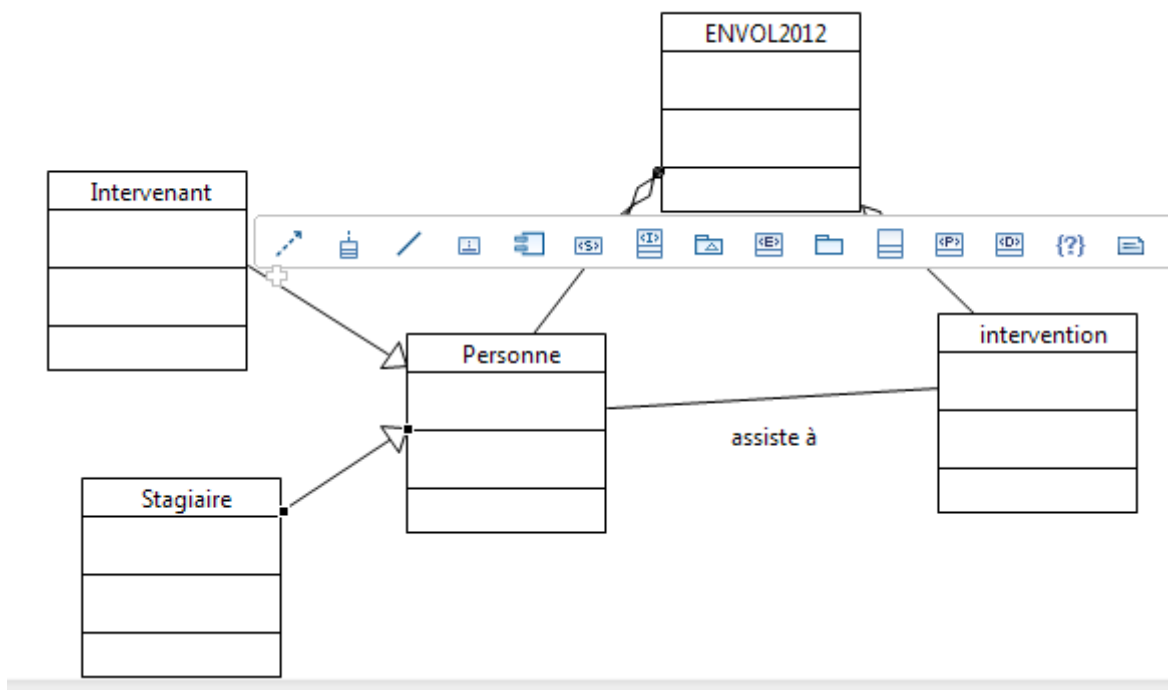
Penser à cliquer sur le petit triangle devant stagiaire7318 pour dérouler les éléments. Nous constatons que nous retrouvons une image similaire à notre fenêtre Project Explorer initiale. Retournons maintenant dans la perspective papyrus A vous maintenant de retrouver comment faire ;-)

Nous allons maintenant modifier notre modèle initial, pour spécifier par exemple qu'il y a deux types de personnes : Intervenant et stagiaire. En UML, on utilise pour cela le symbole de généralisation/spécialisation qui se représente comme si s'était une grosse flèche (oui cela peut prêter à confusion ... mais bon c'est la norme).

Dans un premier temps, créer les deux classes intervenant et stagiaire. Puis sélectionner le lien Generalization, et enfin cliquer d'abord sur la classe Intervenant puis Personne. Faire de même pour stagiaire.



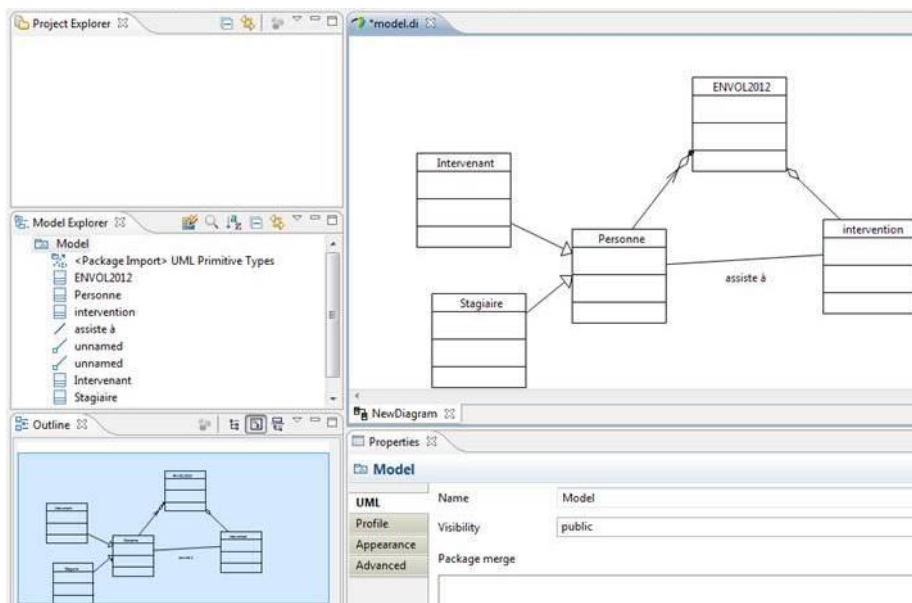
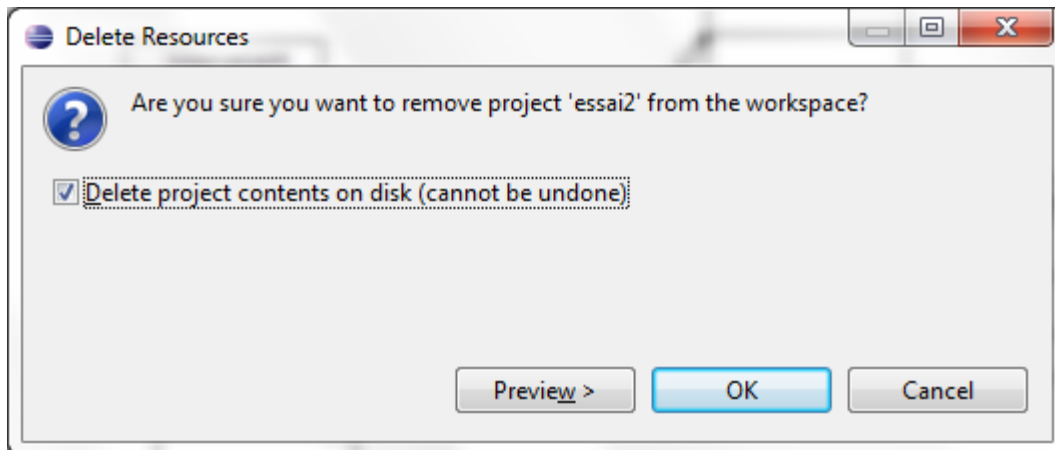
Nous obtenons :



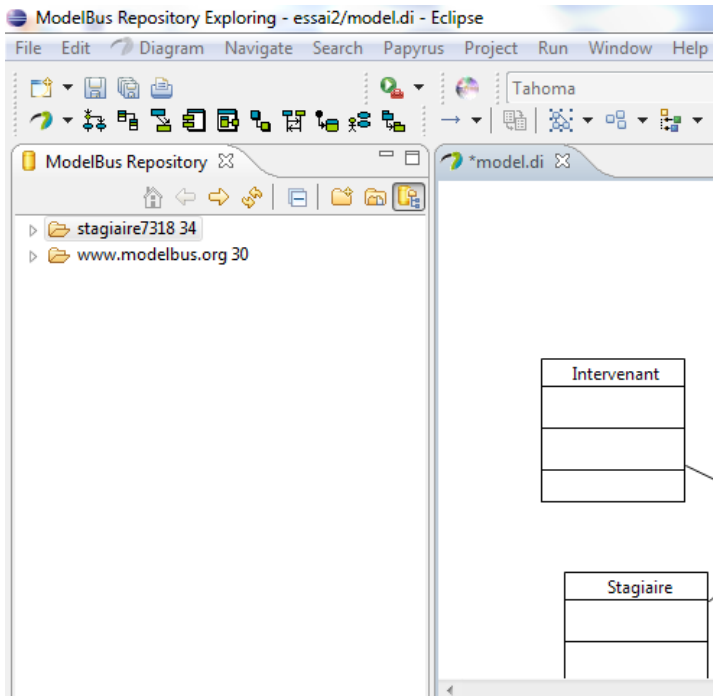
Sauvegarder le projet !

Nous avons encore une fois un projet local différent du repository. Mauvaise nouvelle, nous voulons revenir en arrière sans les deux dernières classes. Une première solution pourrait-être de supprimer les deux dernières classes avec les généralisations/spécialisations associées. Mais, on peut aussi par exemple s'appuyer sur le repository, qui lui est correcte. Soyons fou, supprimons le projet essai en local.

Dans project Explorer, faire un cliquer droit sur le projet essai, et choisir le menu delete. Cocher la case « Delete project contents ... ».

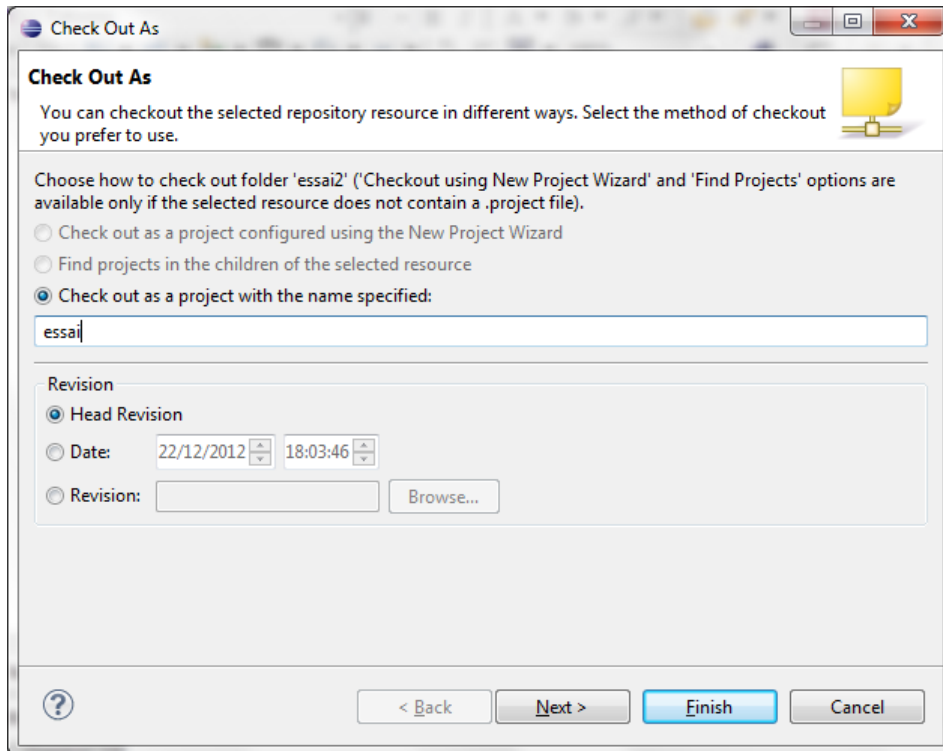


On observe dans Project Explorer que le projet n'apparaît plus. Mais on conserve des « restes » purement visuels car le projet n'existe plus : model.di, model Explorer et outline. Ceci n'est pas très important et n'aura pas de conséquence sur la suite bien sûr. Changeons de perspective en choisissant « ModelBus Repository Exploring ».

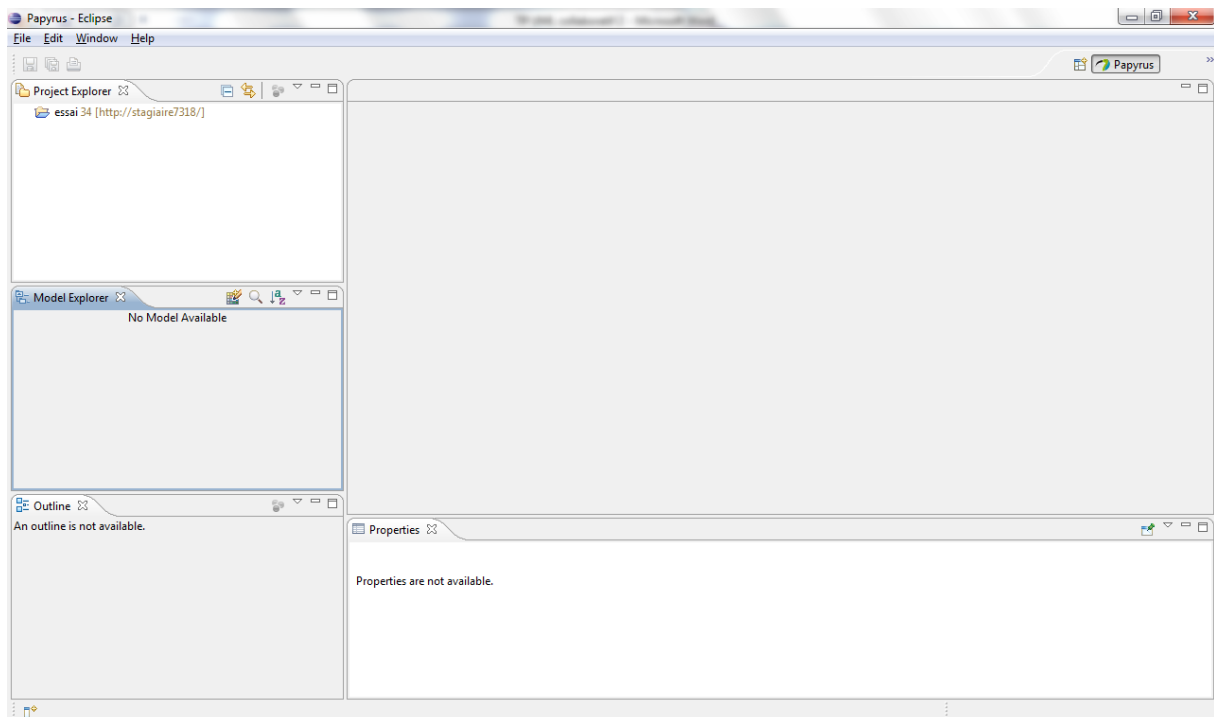


Il ne nous reste plus que model.di comme « verrue visuelle », on le supprime en cliquant sur la croix blanche à côté du nom. Ignorer ensuite la proposition de sauvegarde.

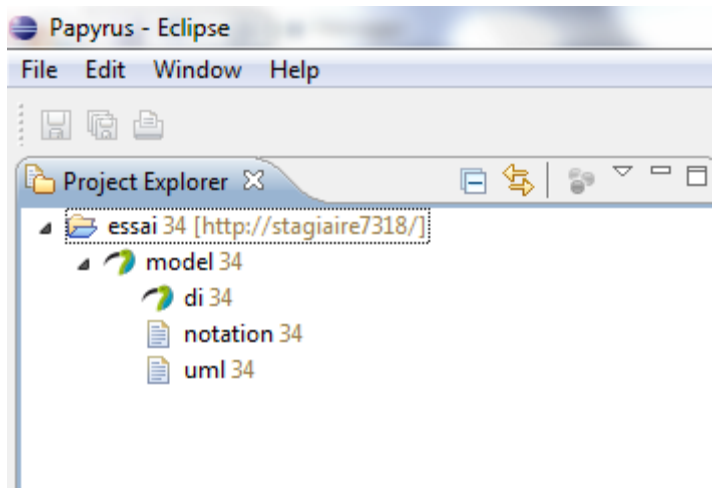
On va ensuite copier le projet stagiaire7318 du repository dans notre dossier local, en faisant un clic droit dessus puis en sélectionnant Find/Check Out As.



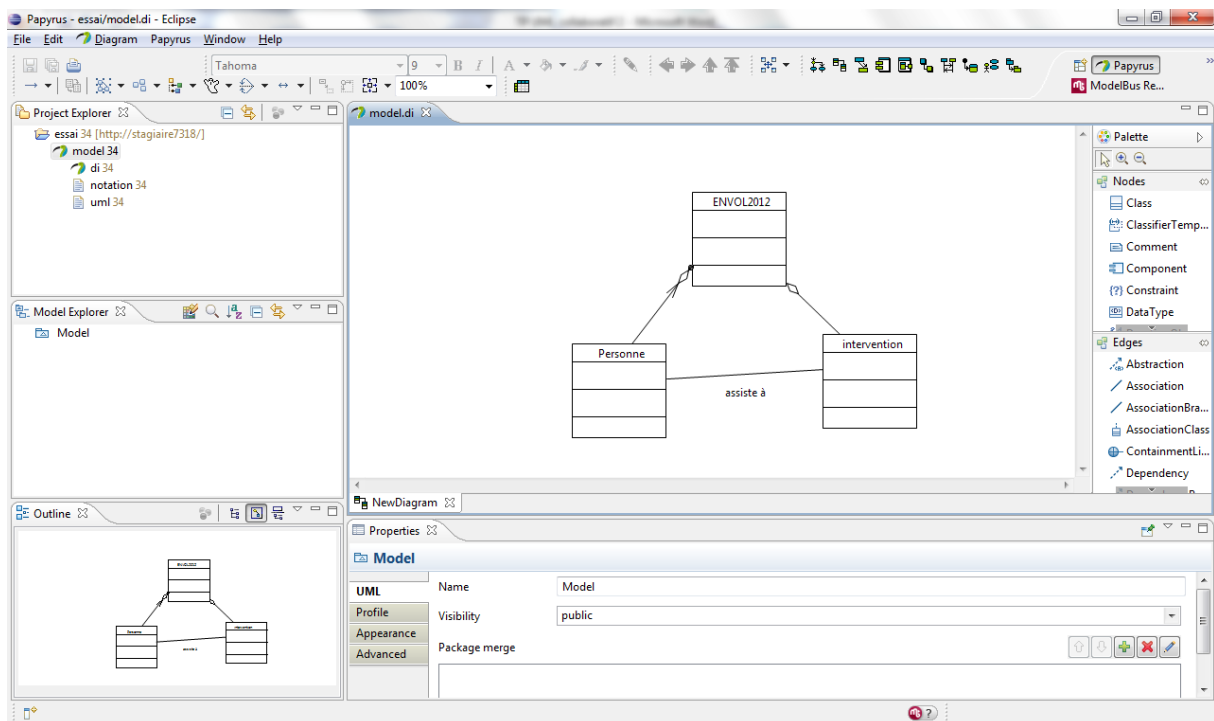
Ici on peut changer le nom par celui que l'on souhaite utiliser en local, voir reprendre une version antérieure car le repository conserve de manière séparée toutes nos sauvegardes dans des versions (révisions). Dans le cadre de ce TP nous n'explorerons pas ces possibilités, donc on valide par finish.



Pas grand-chose de visible, sauf le nom du projet : essai. On clique sur le triangle à sa gauche pour faire apparaître ses éléments.



Puis on fait un double clique gauche sur model.



On retrouve nos petits. Fermons notre projet : clique droit sur essai, puis Close Project.