

# Génie logiciel avancé - 2016

## TD n°4

### Système de planification des vols

---

La présentation rendue les semaines précédentes doit être complétée par toutes les informations nécessaires à la bonne compréhension du projet par un intervenant externe.

L'objectif de ce TD est de compléter ce livrable et de continuer la mise en place votre environnement de développement pour le dimanche 21 février au plus tard.

#### Question 1 – Planification de projet

Tel qu'expliqué en cours, la séance de TD n°6 sera dédiée à des soutenances intermédiaires vous permettant de présenter tout ou partie du projet.

Afin de vous permettre de préparer efficacement cette soutenance, vous devez vous mettre d'accord avec votre chargé de TD sur l'état d'avancement attendu au bout de 6 semaines.

A minima, vous devrez présenter les éléments suivants :

1. Une présentation à l'oral s'appuyant sur le livrable préparé au cours de TD. Il peut bien entendu être mis à jour et amélioré ;
2. La prise de parole n'est pas obligatoire pour tous les membres du groupe ;
3. La présence de l'ensemble du groupe est obligatoire, des questions pouvant être posées à chacun ;
4. La présentation peut être faite en anglais ;
5. Une démonstration d'une partie de l'application avec au minimum une des deux interfaces graphiques.

## Question 2 – Bouchons, seconde partie

Vous avez réalisé lors du TD précédent un bouchon pour les web-services.

De la même manière, vous devez en réaliser un pour la couche de persistance. **Ce bouchon ne doit pas utiliser DataNucleus !** Il doit implémenter les interfaces que vous avez définies durant le TP précédent.

Afin de faciliter son intégration, vous pouvez construire votre DAO autour du design-pattern « Factory » qui permet d'abstraire les implémentations effectives.

Pour réaliser cela, vous devez créer une classe « DAO » que ne contiendra que des méthodes statiques pour accéder aux instances de votre couche de persistance. Un appel au DAO reviendra donc au code suivant :

```
DAO.getActionDao().getActions("user1");
```

L'implémentation de la méthode « getActionDao() » s'appuiera donc sur le bouchon le temps de réaliser la vraie implémentation de la couche de persistance :

```
1 package com.example.datanucleus.dao;
2
3 import com.example.datanucleus.dao.fake.ActionDaoFakeImpl;
4
5 public class DAO {
6
7     public static ActionDao getActionDao() {
8         return new ActionDaoFakeImpl();
9     }
10
11 }
12
```

### Question 3 – Mise en place DataNucleus

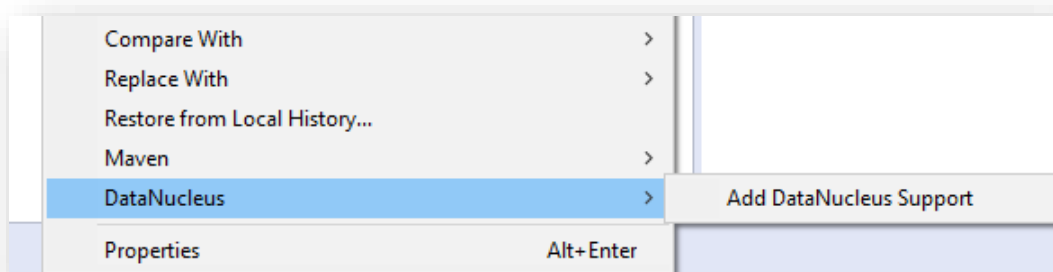
Une fois le bouchon réalisé, il faut mettre en place l'environnement qui vous permettra de réaliser la vraie implémentation. Il vous faut donc utiliser DataNucleus.

Pour cela, le plus simple est d'installer le plugin Eclipse qui se chargera de lancer automatiquement la fonction « Enhance ». Pour l'installer, il vous suffit de suivre les instructions présentes sur la page suivante :

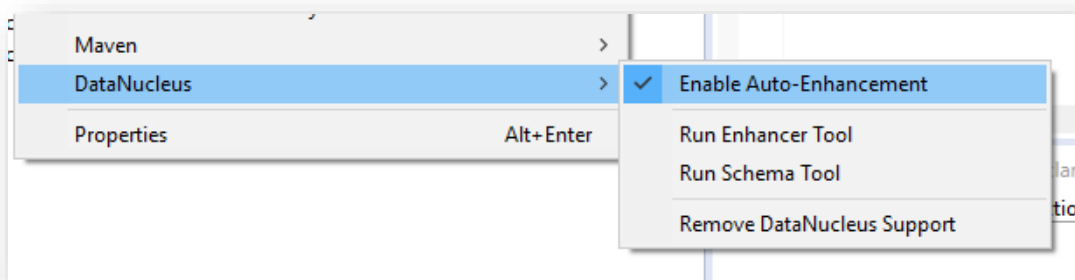
[http://www.datanucleus.org/products/accessplatform\\_4\\_2/jdo/guides/eclipse.html](http://www.datanucleus.org/products/accessplatform_4_2/jdo/guides/eclipse.html)

Une fois installé, le plugin doit être activé en mode automatique sur le projet concerné. Pour cela, vous devez :

1. Ajouter le support « DataNucleus » sur le projet



2. Activer le lancement automatique



Pour débiter, vous pouvez récupérer via GitHub un projet Java qui met en place DataNucleus et HSQLDB via l'url suivante : <https://github.com/ktekkal/datanucleus>