



Initialisation



- Lancer la VM
 - login : **siad** - password : **javaeesiad**
- Ouvrez une console et rendez vous dans le repertoire de dev
 - Créez un repertoire workspaces et placez vous dedans
 - Astuce : `mkdir workspaces && cd $_`
- Il s'agit maintenant de cloner le repository de source (hébergé par github)
 - Exécutez `git clone https://github.com/ptitbob/siad_m2_ws.git`
 - Placez vous dans le repertoire `siad_m2_ws`
 - Exécutez `mvn clean compile` (il devrait télécharger le web)

Pour faire les mise à jour pour chaque cours (si la VM n'a pas été réinitialisée) :

- `git pull origin`
- `mvn clean compile`

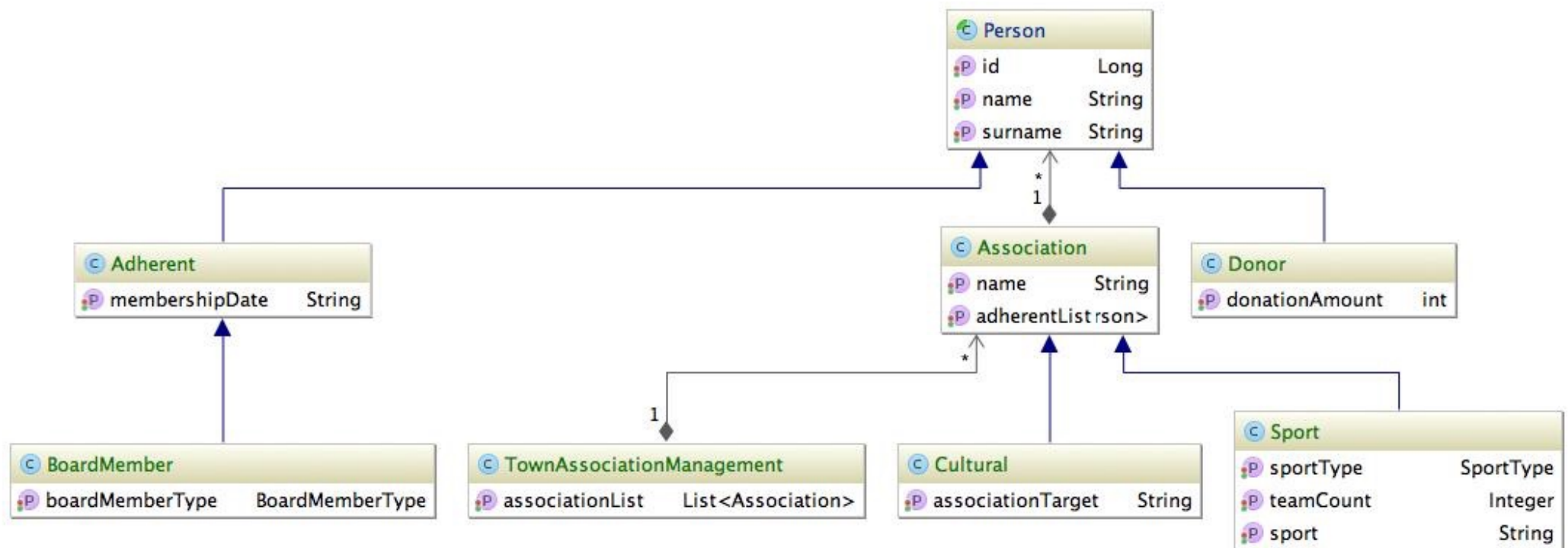
Les outils du cours

- VM Linux (xubuntu)
- Java 8
- IDEs
 - IntelliJ
 - Netbeans
- Maven
- Serveur d'application
 - Glassfish
 - Wildfly
 - Jetty (container de servlet)
- Git
- shell



maven

JAXB - TD 1



Powered by yFiles

Rappelez vous :

- un bean a toujours un constructeur vide
- Le code est en anglais
- les commentaires et javadoc en français

JAXB - TD 1



A partir de la classe de test `TownAssociationTest` et de la méthode principale...

- Suivez les demandes exprimées en commentaire
 - Dans la classe de test et dans chaque bean
- Sur la classe `Person`
 - Ajouter un `id` qui sera obligatoirement en attribut
- Sur la classe `BoardMember`
 - Le type de poste devra être en attribut
- La classe association devra présenter un champs nombre d'adhérents non sérialisé
- Créer une classe `Address`, chaque personne devra la présenter
- Créer un schéma pour chaque package avec comme namespace
 - base (`fr.univ.blois.siad.m2.ws.jaxb.td01`) : **town**
 - `Person` : **adh**
 - `Club` : **clb**
- Pour l'énum `SportType`, il devra être serialisé sous forme d'entier
- Générer le fichier
- *Si il vous reste du temps, amusez vous à générer les schémas*

SOAP Initialisation



- Pour ce TD, nous allons utiliser NetBeans
 - Utilisation du serveur Glassfish intégré
 - Utilisation du serveur de base de données Derby
- Récupérer les dernières sources du projet
 - cf. instruction d'initialisation
 - mvn clean compile
- Il convient de l'initialiser
 - Lancer Netbeans
 - Création de la base
 - Lancer le serveur Derby
 - créer une base activity
 - user : root
 - password : root
 - Importer le projet siad_m2_ws
 - ouvrir le module soap
 - le lancer
 - Voir la page d'index qui s'affiche

TD 2



- Avec la classe PingWS
 - La transformer en VWS
 - Ajouter une méthode qui renvoi une chaine en majuscule
- Créer un service Web à destination des utilisateur
 - Création d'un utilisateur (qui renvoie l'utilisateur créé)
 - Liste des utilisateurs
- Créer un service Web à destination des activités
 - Renvoyer la liste des activités
 - Pour une activité
 - Renvoyer la liste des adhésions
 - Ajouter une adhésion pour un utilisateur

Evidement, pour tout cela, il convient de tester les méthodes...