

TD/TP n° 4

Pour le TP : on utilisera le logiciel Umbrello, sous Linux. À l'issue du TP, la production devra être envoyée sur EPREL : M3105-CP00 > Travaux > Soumission du TP n° 4...

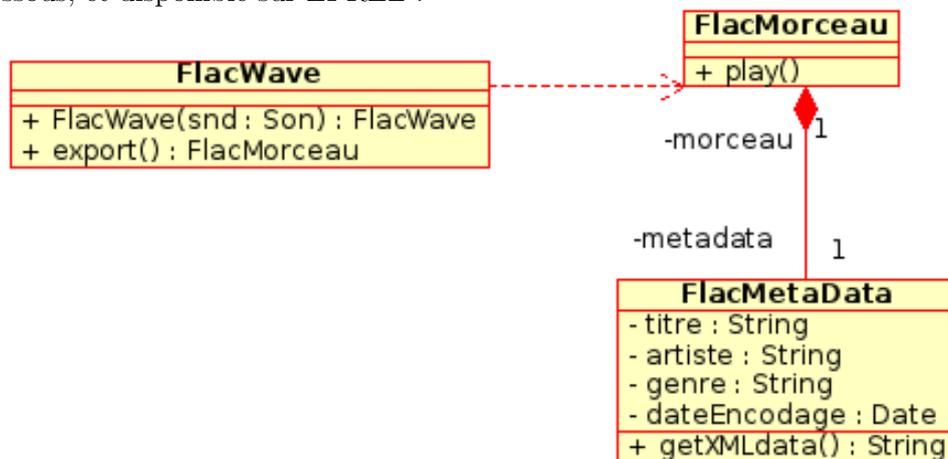
On commencera par les exercices 6 et 7 de la feuille de TD n° 3.

Exercice 1. Un système de reconnaissance musicale

On souhaite programmer un système de reconnaissance musicale de la manière suivante. Le système peut lire un son qui se trouve sur un média externe (lecteur de vinyle, de cassette, de CD...). Puisque le lecteur de vinyle est à la fois externe et peu fiables, on veut que le système gère l'absence de connexion de ce lecteur.

Les sons sont extraits sous forme abstraite (qu'on ne détaillera pas), et peuvent ensuite être exportés de diverses manières : sous la forme d'une partition, sous la forme d'un son MIDI, ou sous la forme d'un son FLAC. On envisage d'enrichir ces formats d'export en fonction des besoins et des formats musicaux qui seraient ultérieurement définis. Les sons exportés seront gardés dans le système, horodatés de la date d'export (`dateExport`) et autre métadonnées (`titre`, `artiste`, `genre`).

Pour l'export d'un son en FLAC, on utilise une bibliothèque externe dont le diagramme de classe est représenté ci-dessous, et disponible sur EPREL :



Les exports dans les autres formats n'étant pas encore implémentés, on peut choisir la forme qu'auront leurs classes. Cependant, on veut dans tous les cas (même pour une partition) qu'un morceau exporté puisse être joué par une opération `play`. De plus, les métadonnées devront pouvoir être exportées en XML par l'opération `getXMLdata`.

1. Construire dans un diagramme à part les classes (vides) `Son` et `Date`. Elles serviront dans la suite mais nous ne les détaillerons pas.
2. Sur un nouveau diagramme, on souhaite modéliser la partie du système qui s'occupe de l'import de vinyles.
 - a) Quel patron de conception vu en cours est utile dans ce cas ?
 - b) Construire le diagramme concernant le vinyle.
3. Sur un autre diagramme, construire le diagramme concernant l'import de sons en général. On remarque que l'aspect crucial et commun aux différentes manières d'importer du son est qu'ils sont capables de fournir un son.
4. Enfin, sur un dernier diagramme, on veut se charger de l'export des morceaux.
 - a) On nous fournit des classes dont la signature des fonctions n'est pas parfaitement en adéquation avec ce que l'on attend. Quels patrons de conceptions vus en cours répondent à ce besoin ?
 - b) D'autre part, on a besoin de classes qui fabriquent toutes des `Morceaux` de musique, mais sous des formes différentes. Puisque l'on ne s'interdit pas d'ajouter de nouveaux formats, que faut-il abstraire ? Quel patron serait utile ici ?
 - c) Construire le diagramme concernant l'export de morceaux.

Exercice 2. Une base de donnée généalogique

On souhaite stocker des informations sur la généalogie d'une (ou plusieurs) familles. Toute personne a un nom et un prénom, ainsi qu'une date de naissance approximative. Elle a au plus deux parents (un père et une mère) et un nombre arbitraire de conjoints et d'enfants.

La base de donnée pouvant être très grande, on ne veut pas que la construction d'un objet `Personne` entraîne l'import depuis la base de donnée de tous ceux qui lui sont reliés à tous les degrés : on ne souhaite récupérer que les informations (nom, prénom, date de naissance approximative) des voisins directs.

Par ailleurs, toutes les requêtes à la base de donnée passent par un objet de classe `BaseDeDonnées`, qui lit directement le SQL par sa méthode `execute(req: String)`.

1. Construire le diagramme de classe d'un système qui ne charge les informations sur les personnes que lorsqu'on y accède réellement. On pourra commencer par répondre aux questions suivantes :
 - a) Quel patron aiderait à modéliser la structure d'un *arbre* généalogique ?
 - b) Quel patron permet de ne pas charger toutes les informations d'un objet ?
2. Donner dans une note le (pseudo-)code Java de l'affichage de toute la lignée maternelle d'une personne. Cette lignée s'arrête si une personne ne dispose pas de mère dans la base de donnée.