Spring Data JPA

Spring Data JPA is an abstraction on top of JPA and hibernate is specification for accessing persisting managing data between java object, hibernate is one implementation of this interface

Hibernate take any java object right her

JAXB est l'acronyme de Java Architecture for XML Binding.

JAXB est une spécification qui permet de faire correspondre un document XML à un ensemble de classes et vice versa au moyen d'opérations de sérialisation/désérialisation nommées marshalling/unmarshalling.

JAXB permet aux développeurs de manipuler un document XML sans avoir à connaître XML ou la façon dont un document XML est traité comme cela est le cas avec SAX, DOM ou StAX. La manipulation du document XML se fait en utilisant des objets précédemment générés à partir d'une DTD pour JAXB 1.0 et d'un schéma XML du document à traiter pour JAXB 2.0.

Le but de l'API et des spécifications JAXB est de faciliter la manipulation d'un document XML en générant un ensemble de classes qui fournissent un niveau d'abstraction plus élevé que l'utilisation de JAXP (SAX ou DOM). Avec ces deux API, toute la logique de traitements des données contenues dans le document est à écrire.

L'API JAXB propose un framework composé de classes regroupées dans trois packages :

* javax.xml.bind : contient les interfaces principales et la classe JAXBContext
* javax.xml.bind.util : contient des utilitaires
* javax.xml.bind.helper : contient une implémentation partielle de certaines interfaces pour faciliter le développement d'une implémentation des spécifications de JAXB

47.1.3. L'utilisation des classes générées et de l'API

Pour pouvoir utiliser JAXP, il faut tout d'abord obtenir un objet de type JAXBContext qui est le point d'entrée pour utiliser l'API. Il faut ensuite utiliser la méthode newInstance() qui attend en paramètre le nom du package contennant les interfaces générées (celui fournit au paramètre -p de la commande xjc).

Pour pouvoir créer en mémoire les objets qui représentent le document XML, il faut à partir de l'instance du type JAXBContext, appeler la méthode createUnmarshaller() qui renvoie un objet de type Unmarshaller.

L'appel de la méthode unmarshal() permet de créer les différents objets.

Pour parcourir le document, il suffit d'utiliser les différents objets instanciés.

### La création d'un nouveau document XML

Parmi les classes générées à partir du schéma XML, il y a la classe ObjectFactory qui permet de créer des instances des autres classes générées.

XSD(*XML Schema Definition)*

**XML Schema**, publié comme recommandation par le [W3C](https://fr.wikipedia.org/wiki/W3C) en [mai 2001](https://fr.wikipedia.org/wiki/Mai_2001), est un [langage de description de format de document](https://fr.wikipedia.org/wiki/Langage_de_description_de_format_de_document) [XML](https://fr.wikipedia.org/wiki/Extensible_markup_language) permettant de définir la structure et le type de contenu d'un document XML. Cette définition permet notamment de vérifier la validité de ce document.

Une instance d'un *XML Schema* est un peu l'équivalent d'une *définition de type de document* ([DTD](https://fr.wikipedia.org/wiki/Document_Type_Definition)). XML Schema amène cependant plusieurs différences avec les DTD : il permet par exemple de définir des domaines de validité pour la valeur d'un champ, alors que cela n'est pas possible dans une DTD ; en revanche, il ne permet pas de définir des [entités](https://fr.wikipedia.org/wiki/Entit%C3%A9_de_caract%C3%A8re) ; XML Schema est lui-même un document XML, alors que les DTD sont des documents [SGML](https://fr.wikipedia.org/wiki/SGML).

Ce langage de description de contenu de documents XML est lui-même défini par un schéma, dont les [balises](https://fr.wikipedia.org/wiki/Langage_de_balisage) de définition s'auto-définissent (c'est un exemple de [définition récursive](https://fr.wikipedia.org/wiki/R%C3%A9cursivit%C3%A9))[1](https://fr.wikipedia.org/wiki/XML_Schema#cite_note-1)

La recommandation spécifie la validation des documents XML à partir d'un modèle abstrait ; elle en fournit le format XML.

Pour ce modèle abstrait, un schéma est un ensemble de composants, tels que :

* la déclaration d'éléments (la notion d'éléments vient de la recommandation XML),
* la déclaration d'attributs (qui vient aussi de la recommandation XML),
* la définition de types simples (valeurs constituées uniquement à partir d'une chaîne de caractères),
* la définition de types complexes (valeurs constituées d'attributs et d'autres valeurs).

Elle introduit également le type *anyType* (n'importe quel type), base de tous les types utilisés.