Data Modeling
Exercise

TABLE OF CONTENTS

GOAL

This exercise covers the basics of data modeling in SAP Commerce Cloud. In this exercise, you will learn how to build onto an already-defined data model by defining additional types and relations.

INSTRUCTIONS

First, take a look at the UML diagram below. This is the data model you will be working with. The colors represent the status of a data type (item type) in the system.

If you can't see the colors, the OOTB items (Product and User) are in dark-bordered rectangles, and items you will need to define are in bold text.

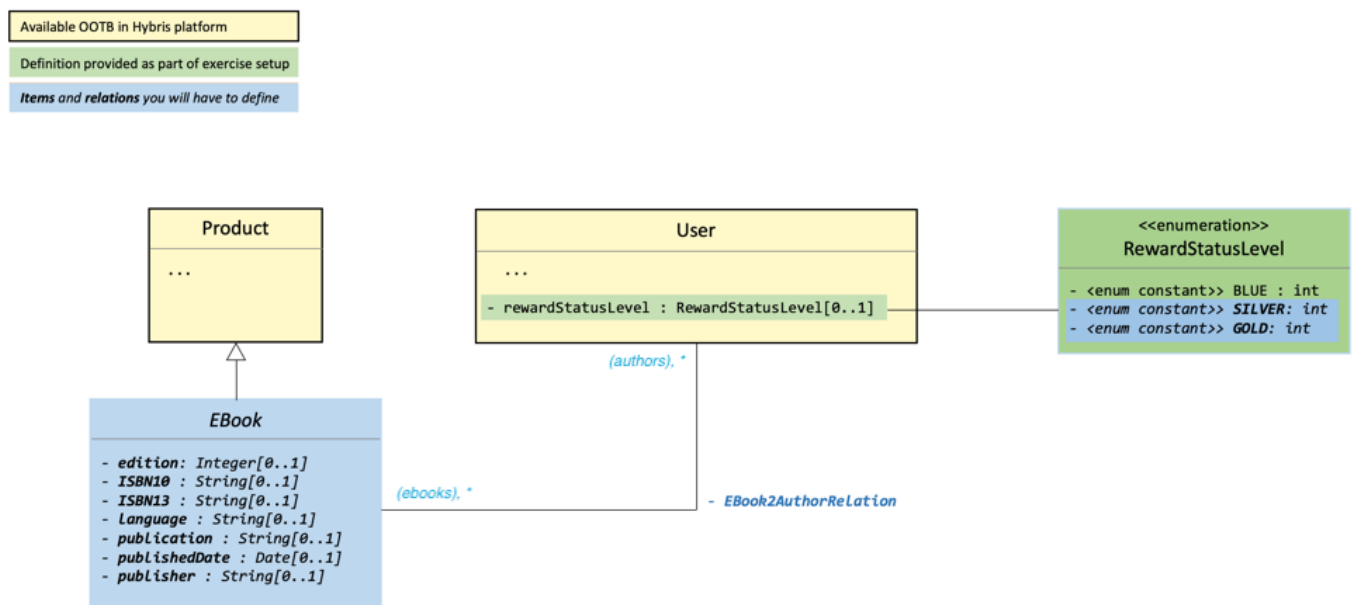|	
	the corresponding item is available OOTB in the SAP Commerce Cloud platform.
	a definition of this item or property is provided as part of the exercise setup.
	Items, **properties**, and **relations** you will have to define (denoted in bold type).

Figure: UML Diagram of Data Model

Preparation

P1   To prepare your system for working on this exercise, open a *Terminal* or *cmd* window, navigate to the *MYPATH*/workspace/TrainingLabTools/exercise_DataModeling directory, and run the following Ant target:

```
ant –f data_modeling_tasks.xml setup
```

This will create a new extension named trainingdatamodeling and add it to your localextensions.xml.

P2   Now you have to import the new extension into your IDE. In Eclipse, click: *File > Import*. In the dialog, select *General > Existing Projects into Workspace*. In the **Import Projects** dialog, select *MYPATH*/workspace/hybris/bin/custom/trainingdatamodeling as the root directory and click **FINISH**. The new extension is now visible in your IDE.

Now you can start working on the exercise. You will be editing two files, `trainingdatamodeling-items.xml` and `trainingdatamodeling-locales_en.properties`, located here:

`MYPATH/workspace/hybris/bin/custom/trainingdatamodeling/resources/trainingdatamodeling-items.xml`

`MYPATH/workspacee/hybris/bin/custom/trainingdatamodeling/resources/localization/trainingdatamodeling-locales_en.properties`

Step 1 - Define the EBook type and add EBook Attributes

Using the UML diagram as a reference, complete the definition of the *EBook* item type in `trainingdatamodeling-items.xml` (which was copied to your `trainingdatamodeling` extension's resources folder by the exercise's *ant* setup task). As noted above, the attributes you need to add are highlighted in blue and bold.

> **HINTS**
>
> 1) EBook is a subtype of Product.
> 2) The UML diagram uses an array-style notation to indicate an attribute's cardinality (i.e., [1] denotes an attribute for which a value is mandatory; [0..1] denotes an attribute for which a value is optional). Each endpoint of a relation indicates that attribute's cardinality with either 1 or * (many). Look at other attributes of the EBook item type in trainingdatamodeling-items.xml as a reference.
> 3) When declaring an attribute, don't forget the mandatory tag: <persistence type="…" />.

Step 2 - Add a Relation

Create a many-to-many relation between EBook and User as shown in the UML diagram. Use **EBook2AuthorRelation** as the code for this relation (please enter the code exactly as specified, including the capitalization; if you don't, the verification script will complain). Use **ebooks** and **authors**, respectively, as qualifiers for EBook and User in this relation. To see examples, you can look at relations defined in other extensions' `items.xml` files. (Remember to include the deployment tag required by a many-to-many relation). For this exercise, declare the relation with a **localized="false"** attribute.

Step 3 - Add Enumeration Attributes

The Customer can have 3 different status levels: BLUE, SILVER, and GOLD. Complete the enumType `RewardStatusLevel` defined in `trainingdatamodeling-items.xml`, by adding the missing `<value…` elements. Use the UML diagram as a reference.

Step 4 - Localize Attribute Names

Provide a localized name and description for the `publication` and `publisher` attributes of EBook in `trainingdatamodeling-locales_en.properties`

> **HINTS**
>
> Be careful… you're localizing the **publication** and **publisher** property's meta-data, **not** making each a "localized:" property. If the distinction isn't clear, consult the slides for this chapter or ask your instructor for clarification

Step 5 - Update SAP Commerce Cloud

In the command line, navigate to the platform directory and run `ant all`. Then restart the SAP Commerce Cloud server.

Go to HAC on your browser (http://localhost:9001 or https://localhost:9002) and update the platform (NOTE: you need to check only two options: **Update running system** and **Localize types)**.

> ⚠️ **Update Options**
>
> If you don't know why you should choose these options, please take a look at Initializing and Updating the SAP Commerce Cloud before proceeding.

> ✅ **View Your Changes in Backoffice**
>
> If you are curious, you may view some of the changes that you made to an already-existing type, like **User**, in Backoffice. To do so, login to Backoffice at https://localhost:9002/backoffice/login.zul using the following credentials: (user) `admin`, (password) `nimda`. Then go to *System > Types* and search for the type **User**. Click on **User** and navigate to the Properties tab. The new attributes should appear here. Similarly, you can also search for the **EBook** type, click on it, and navigate to the Properties tab. The new attributes should also appear using the labels you defined in Step 4.

Verify

To verify your solution, go to HAC and run the `verifyDataModelingExercise` script to check your changes.

Solution

If you don't wish to complete this exercise manually, you can install the solution provided:

S1 Navigate to the *Terminal* or *cmd* window where the server is running, and if it is running, stop it by entering `CTRL-C`.

S2 Navigate to `MYPATH/TrainingLabTools/exercise_Data_Modeling` and execute:

```
ant -f data_modeling_tasks.xml solution
```

RECAP

In this exercise you have learned to define new item types and relations based on a data model. You have also learned how you can localize item types and their attributes.

SAP Customer Experience