

## Lesson 9 Subtypes

In this lab, you will use the Entity Editor in Guidewire Studio to modify the TrainingApp data model. You will create a new custom subtype entity. Then, you will extend an existing base application subtype entity.

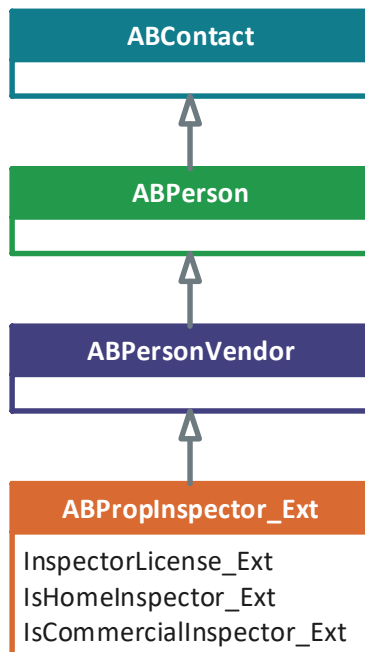
### 9.1 Prerequisites

For this lab, use TrainingApp, Guidewire Studio, and a supported web browser.

<http://localhost:8880/ab/ContactManager.do> is the default URL for TrainingApp. To view, edit, and delete various contacts, log in to TrainingApp as Alice Applegate. The login/password for Alice Applegate is `aapplegate/gw`.

### 9.2 Lab: Create a new custom subtype entity

*“We need to store information about Property Inspectors. After reviewing the data model, we found that there is currently no entity available to do this. Configure the data model and create the new a subtype entity as specified in the diagram and table below.”* – Insurance company business analysts



In this lab, you will create a subtype entity for Property Inspectors using ABPersonVendor as the supertype entity. For each new entity element, remember to set the nullokk attribute to true if not specifically defined to be false. When required, add an element description and specify column parameters.

## 1. Create the **ABPropInspector\_Ext** subtype entity

- a) Create a new ETI file. Specify the following values in the new entity dialog:
  - Entity: **ABPropInspector\_Ext**
  - Entity Type: **subtype**
  - Supertype: **ABPersonVendor**

## 2. In the Entity Editor, add the following fields:

Field Name	Datatype	Null ok?
InspectorLicense_Ext	a string of up to 40 characters	true
IsHomeInspector_Ext	a boolean value	true
IsCommercialInspector_Ext	a boolean value	true

## 1. Use Studio's code generation feature to process the entity and generate the Java class

- a) If possible, use incremental code generation
- b) Verify that there were no errors during code generation

Open the generated Java class and verify that the class extends the ABPersonVendor and you can see the three new properties



### Best Practices

Use **\_Ext** in subtypes name and field names

Guidewire recommends the use of the **\_Ext** suffix (or any unique suffix) in the entity name when creating new (subtype) entities.

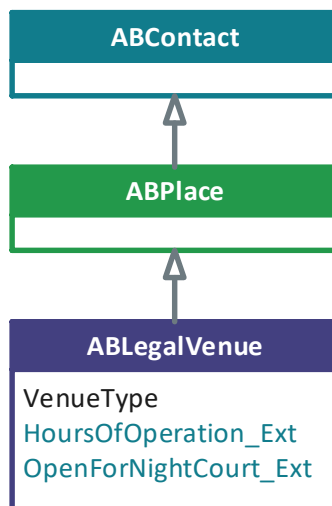
As for fields, it depends on whether the new subtype entity is part of a base application subtype hierarchy or a custom subtype hierarchy.

- The **\_Ext** suffix is recommended if the subtype is part of a base application subtype hierarchy

- The \_Ext suffix is not needed if the subtype is part of a custom subtype hierarchy
- In this lab, ABPersonVendor is the Supertype of ABPropInspector\_Ext. ABPersonVendor is a base application subtype; thus it is part of a base application subtype hierarchy. Guidewire can add new fields to base application subtypes in future releases. These new fields would be inherited and could cause conflict in ABPropInspector\_Ext if the \_Ext (or any unique suffix) is not used in the field names.

## 9.3 Lab: Extend an existing base application subtype entity

*We are going to use the ABLegalVenue OOTB subtype entity to store information about legal venues. However, during our initial analysis, we identified a gap and found that we cannot store the following two pieces of information: (1) the hours of operation and (2) if the legal venue is open for night court or not. Configure the data model and create the new a subtype entity as specified in the diagram and table below.” - Insurance company business analysts*



In this lab, you will create an entity extension of ABLegalVenue, an existing subtype entity. You will create new entity elements to capture specific information for legal venues. For each new entity element, remember to set the nullok attribute to true if not specifically defined to be false. When required, add an element description and specify column parameters.

### 2. Create the ABLegalVenue entity extension

- a) Create ABLegalVenue.etx

**3. In the Entity Editor, add the following fields:**

Field Name	Datatype	Null ok?
HoursOfOperation_Ext	a string of up to 40 characters	true
OpenForNightCourt_Ext	a boolean value	true

**4. Use Studio's code generation feature to process the entity and generate the Java class**

- a) If possible, use incremental code generation
- b) Verify that there were no errors during code generation
- c) Open the generated Java class and verify that you can see the newly added properties



## Best Practices

Use \_Ext in entity field name when extending base application (subtype) entities

For fields that are added to a base application entity, Guidewire recommends that the field name should end with \_Ext (or start with Ext\_). This is to prevent potential conflicts during the upgrade to the next version of the Guidewire application.

### 9.3.1 Verification



## Activity

Verify the work you have done

As a configuration developer, you want to be able to properly deploy new and changed data model resources.

**1. Restart the server to deploy the changes**

- a) During server restart Studio first runs the code generators, then compiles the project and finally deploys the resources

**2. Regenerate the Data Dictionary**

**3. Open the Data Dictionary**

**4. In Windows Explorer, navigate to the data dictionary.**

5. Open the data dictionary using your preferred browser.
6. Verify your changes to the ABPropInspector\_Ext entity
7. Verify your changes to the ABLegalVenue entity



## 9.4 Lab Solution: Create a new custom subtype entity



### Solution

Exact details on how to complete the lab.

1. Create the ABPropInspector\_Ext.eti file in the  
...\configuration\config\Extensions\Entity folder

A screenshot of a software dialog box titled "Entity". It contains several input fields and a checkbox. The "Entity" field is filled with "ABPropInspector\_Ext". The "Entity Type" dropdown menu is set to "subtype". The "Desc" field is empty. The "Supertype" field is filled with "ABPersonVendor" and has a small "..." button to its right. At the bottom, there is a checkbox labeled "Final" which is unchecked, and two buttons labeled "OK" and "Cancel".

Entity

Entity  
ABPropInspector\_Ext

Entity Type  
subtype

Desc  
|

Supertype  
ABPersonVendor

☐ Final

OK Cancel

## 2. Add the fields

ABPropInspector_Ext.eti x				
+ columnParam [Icons] Show All [Icons]				
Element	Primary Value	Secondary Value	Name	
▼ subtype	ABPropInspector...		<b>name</b>	InspectorLicense_Ext
▼ column	InspectorLicense_...	varchar	<b>type</b>	varchar
columnP, size		40	<b>nullok</b>	true

ABPropInspector_Ext.eti x				
+ columnParam [Icons] Show All [Icons]				
Element	Primary Value	Secondary Value	Name	
▼ subtype	ABPropInspector...		<b>name</b>	IsHomeInspector_Ext
▶ column	InspectorLicense_...	varchar	<b>type</b>	bit
column	IsHomeInspector...	bit	<b>nullok</b>	true

ABPropInspector_Ext.eti x				
+ columnParam [Icons] Show All [Icons]				
Element	Primary Value	Secondary Value	Name	
▼ subtype	ABPropInspector...		<b>name</b>	IsCommercialInspector_Ext
▶ column	InspectorLicense_...	varchar	<b>type</b>	bit
column	IsHomeInspector...	bit	<b>nullok</b>	true
column	IsCommercialIns...	bit	desc	