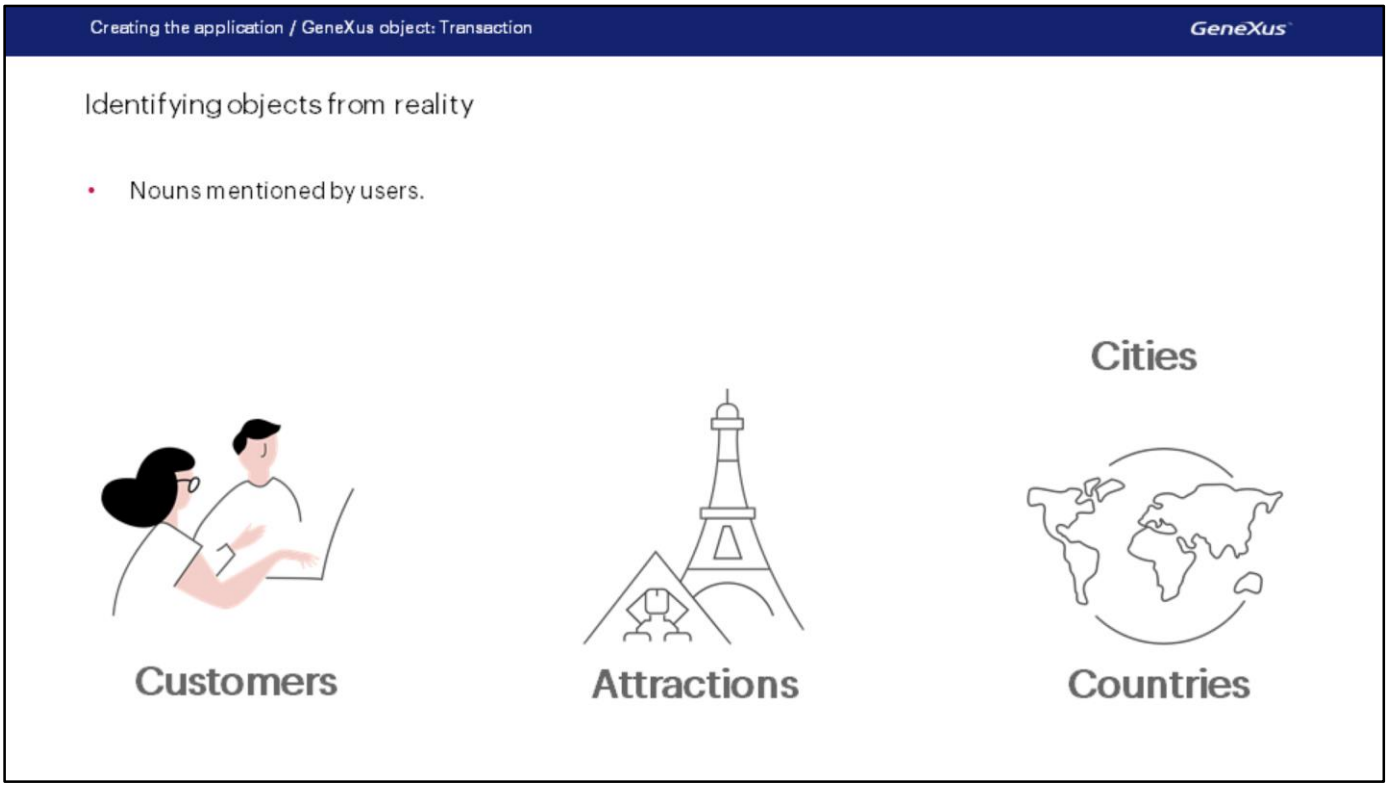
First Transaction Design

Introduction to GeneXus

GeneXus 16



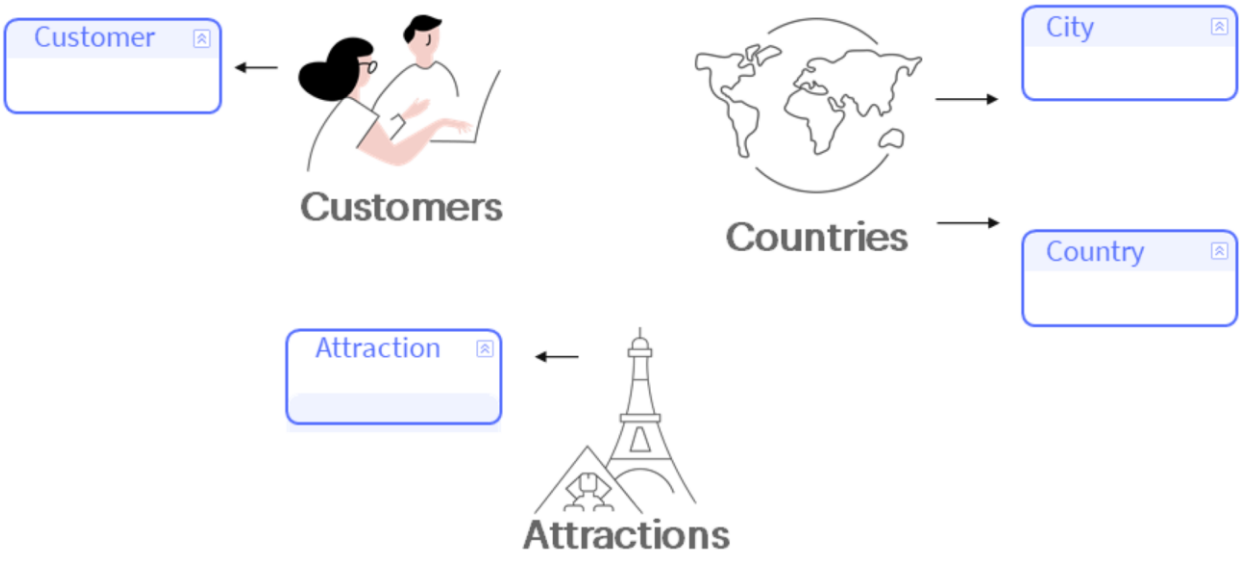
Once the Knowledge Base has been created, the next step is to **describe the objects of reality using GeneXus objects**...

To identify these objects of reality, we recommend paying attention to the words used by the users.

At the travel agency that requested the application we were told that they want to record their **customers**, the **tourist attractions** they recommend, and the **countries** and **cities** they offer tours in.
Based on this information we identify 4 objects of reality to describe in the Knowledge Base:
Clients
Tourist attractions
Countries
Cities

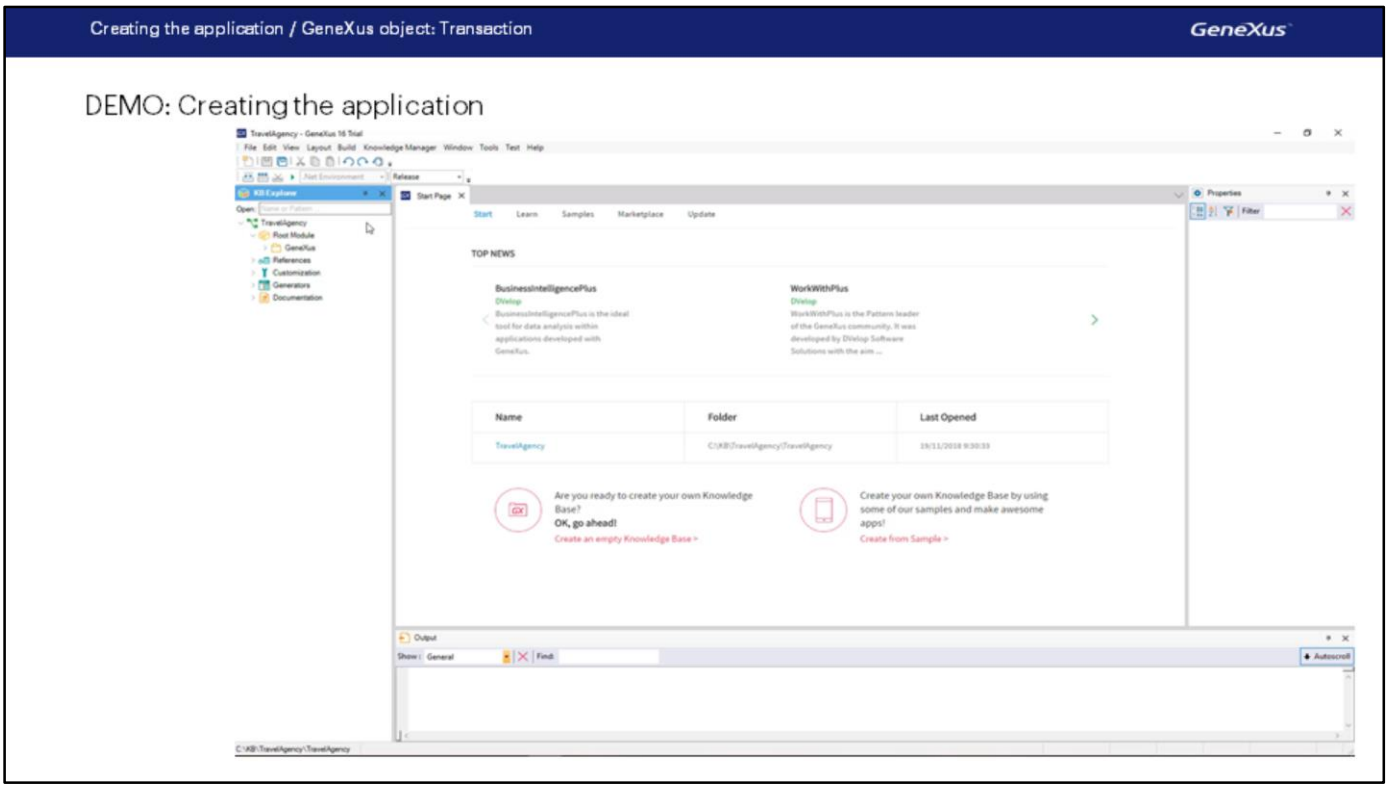
Transactions

- For every object from reality identified we will create a transaction:



For each object of reality that we've identified, **we will create a GeneXus object of transaction type**.

The first GeneXus objects created in a Knowledge Base are transactions, as they allow us to describe objects or actors of reality. Let's go back to GeneXus to do this.



[Demo: https://youtu.be/8jo0FAI4pYI]

To create a GeneXus object we select File / **New** / **Object**

We select this option and see that the following dialog box is displayed to create a GeneXus object, where we can choose the type of object that will be created:

We select the **transaction** object type... and call it “Customer”.

We click on the “Create” **button** ...

Here we can see the transaction that has been created, ready for us to start defining its structure:

All transactions have these sections that we will describe later on.

The transaction structure allows us to select the attributes or fields that describe an object of reality.

In each transaction...

- Define the attributes/fields that describe the object from reality.

Name

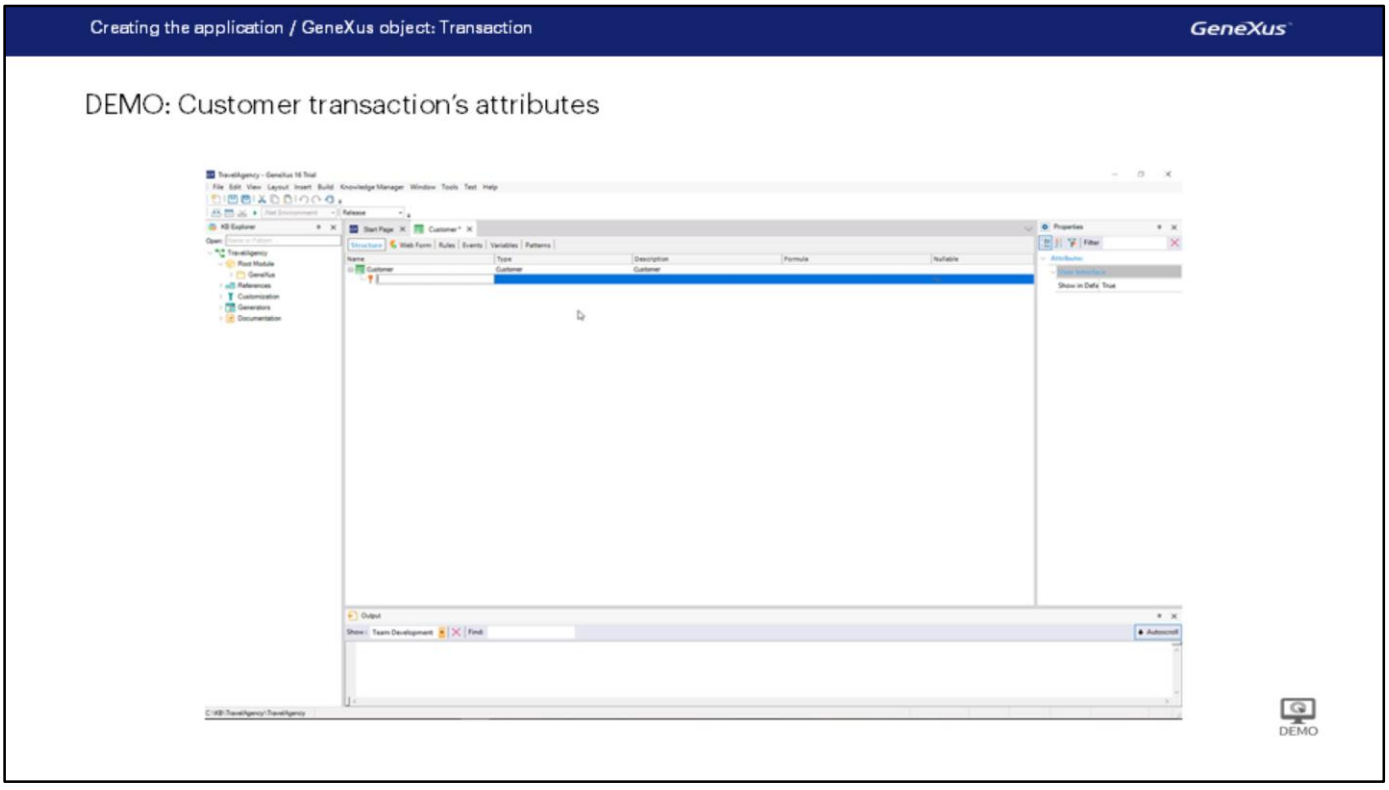
LastName

Address

Phone

Email

At the travel agency we were told that they want to record every customer's **name, surname, address, phone, and email**. Therefore, this data that must be recorded for each customer, **matches the attributes that have to be created for this transaction**.



[Demo: https://youtu.be/5-jCCay2q5Q]

So, let's start creating the **Customer transaction's attributes**.

Note that the first line is created for us to enter the first attribute...
Also, note that an icon key is associated with this line.
The reason is that **in every transaction, an attribute – or set of attributes – must be set with identifier or key role**...
The concept of identifier or key attribute is aimed at uniquely identifying every customer recorded, or any object of reality.

In other words, we will not be able to enter 2 customers with the same identifier value.
We will now set the key attribute of the Customer transaction... since we are not required to save their passport or ID card numbers, which could be candidates for identifier roles, we will create an attribute called "CustomerId".... and soon we will set it to be autonumbered in sequence.
Note that if we press the "dot" key on the keyboard, GeneXus automatically shows the transaction name as prefix in the attribute name...
We only have to type "Id" after the "Customer" prefix:

We press the Tab key ... and choose the **data type** that will be stored for this attribute.

Clicking on the arrow displays the data types available in GeneXus... for this attribute we will leave the default data type, that is to say: **Numeric of 4 digits (with no decimals).**
We press ENTER and start to create the second attribute.

A new line is opened.
Once again, we type "." and complete the attribute name with "Name", that is to say: "CustomerName".

To set the data type stored by the CustomerName attribute, in this case we select the CHARACTER data type.
Note that if we type an opening bracket...
The default length is 20 characters... we will leave it unchanged.

We follow the same steps to enter the CustomerLastName attribute, which will also be of Character type, length 20.

Now we add the CustomerAddress attribute. We can see that in this case the data type was automatically assigned. GeneXus realized that we want to create an attribute whose name partially matches the name of an existing data type.

We continue… with CustomerPhone, and see that GeneXus assigned it the Phone data type.
….. Lastly, we enter the CustomerEmail Attribute, which is assigned the Email data type by GeneXus. In particular, the data types Address, Phone and Email are special data types called **semantic domains**. Later on, we will see that when we work with them, they include features that are specific to an address, a phone number or an email address, respectively.

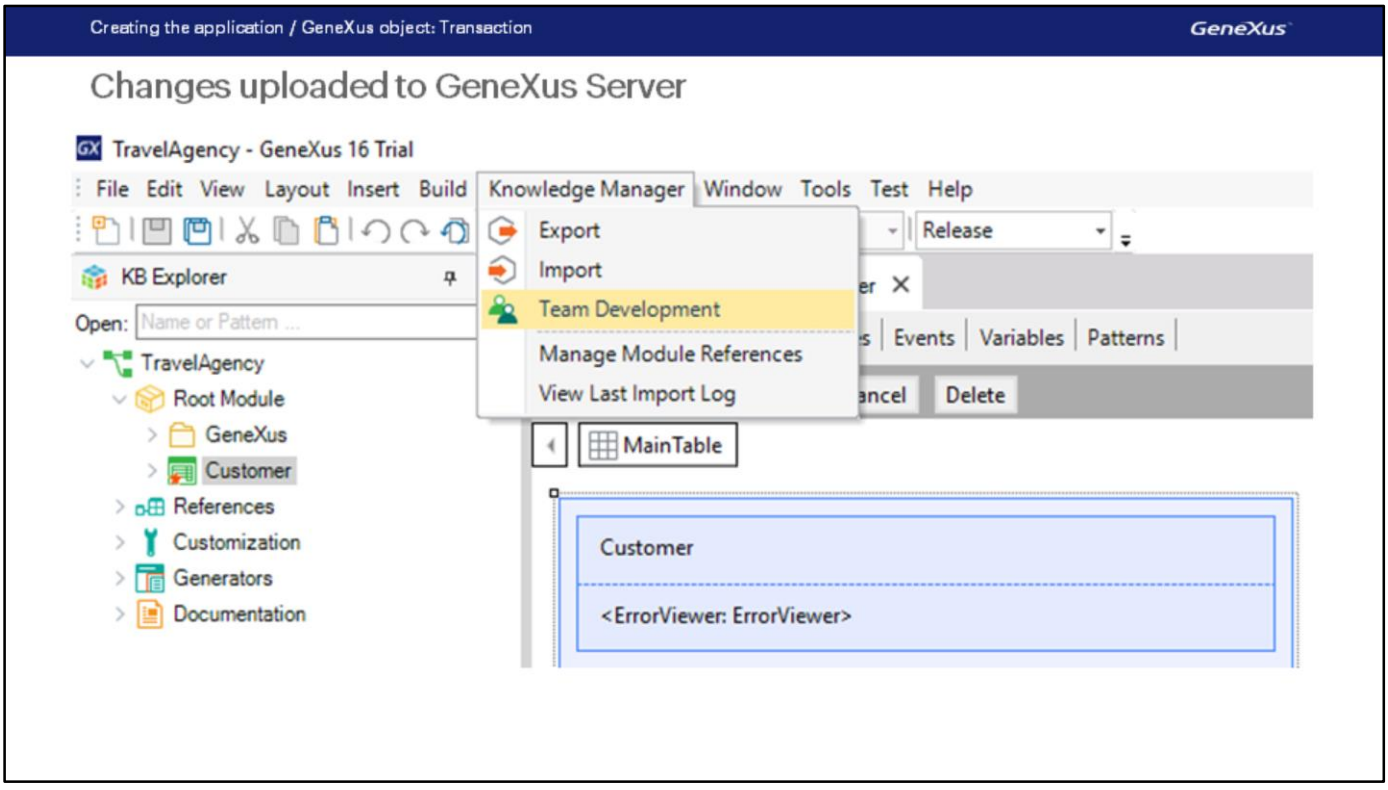
We save this transaction.

Note that **an asterisk is being displayed in the Customer transaction tab...**

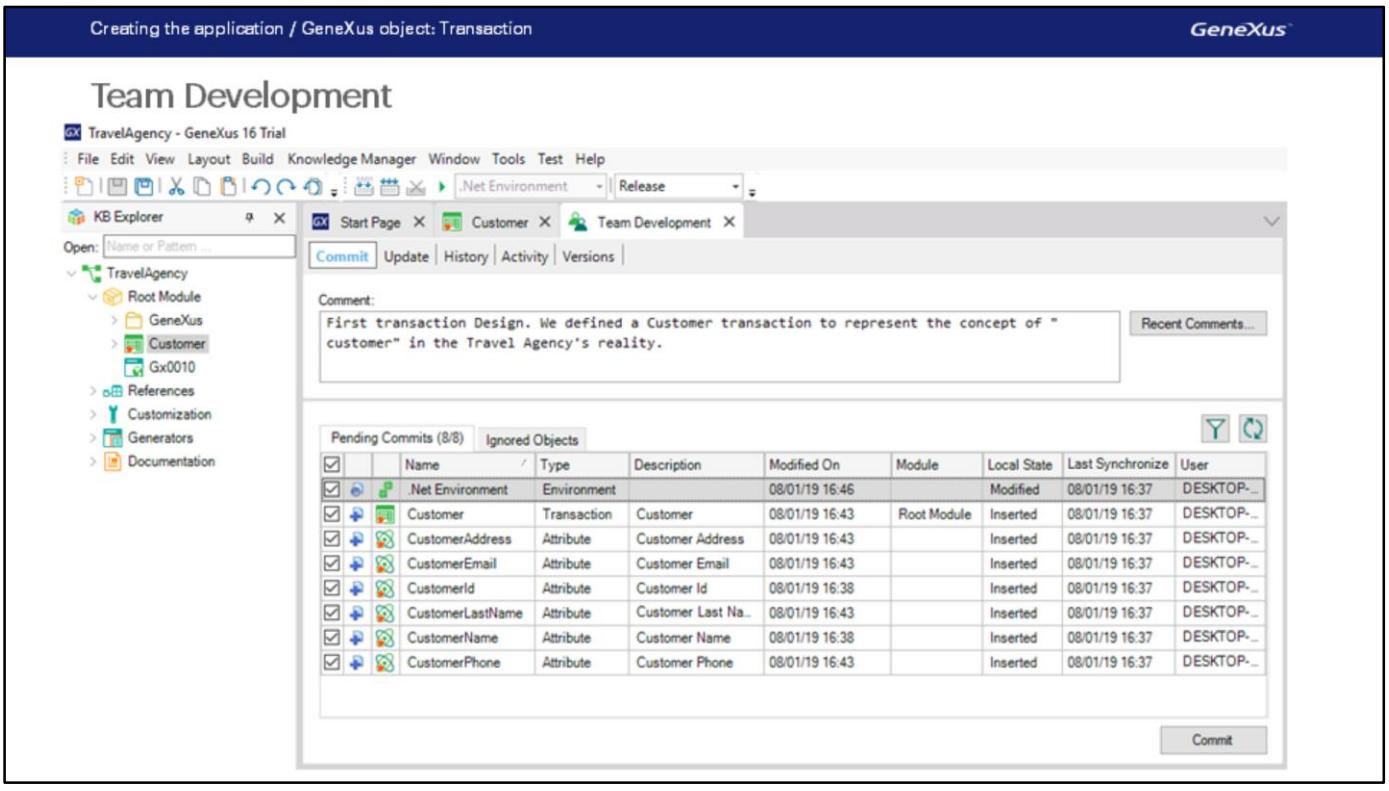
This means that the transaction is being edited... and when we save the changes... The asterisk disappears...

We select the “Web Form” section.

And since ours is a web application, **GeneXus has automatically designed a Web Form according to the defined structure**. This form will allow users to add, change and delete customers.



Now to conclude the job, we will send the changes we made to the KB we uploaded to Gxserver.
To do that we select Knowledge Manager, followed by Team Development.



A window opens up showing all the objects we modified since the last time we saved the KB in the server. Here, we see that the Customer transaction was created, along with this attributes and the Customer table.
Let's add a comment to indicate that…

And we press Commit.

GeneXus™

The power of doing.

Videos	training.genexus.com
Documentation	wiki.genexus.com
Certifications	training.genexus.com/certifications