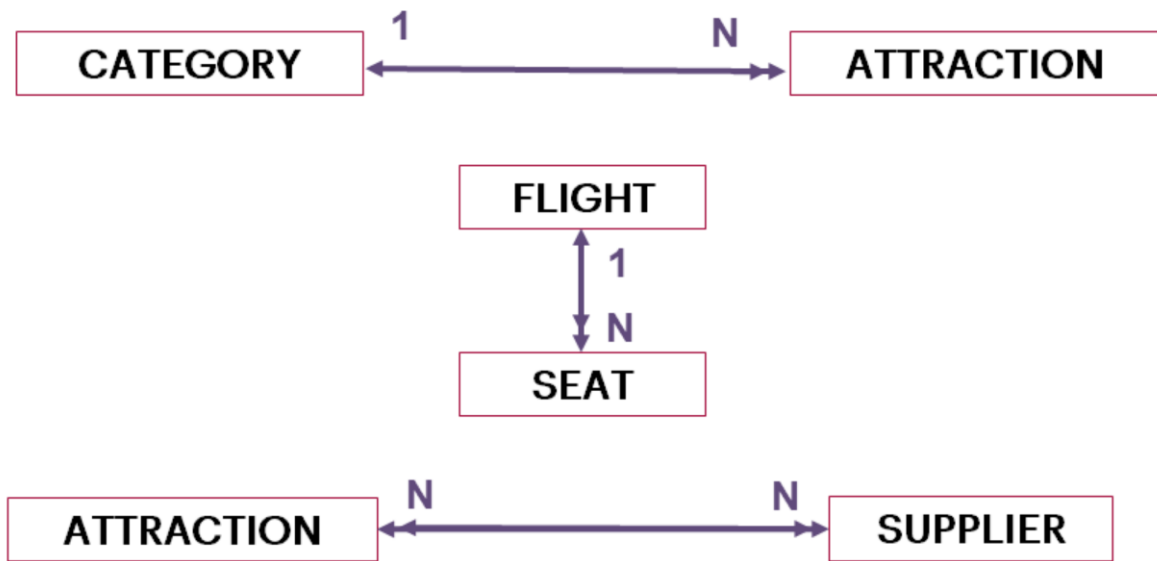
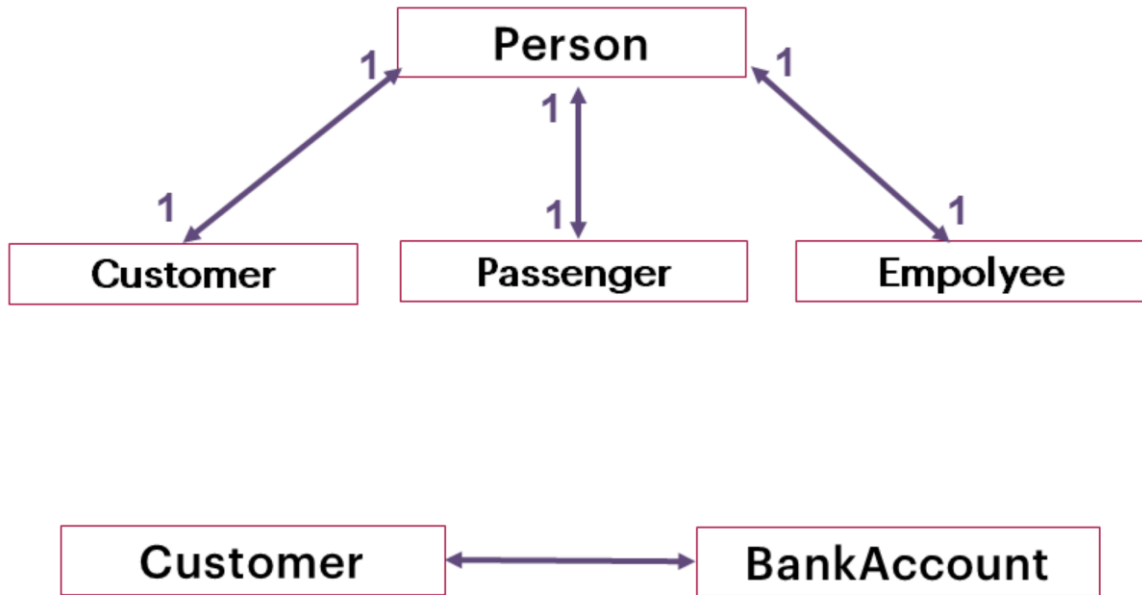


Relaciones 1 a 1 entre actores de la realidad

GeneXus™ 16



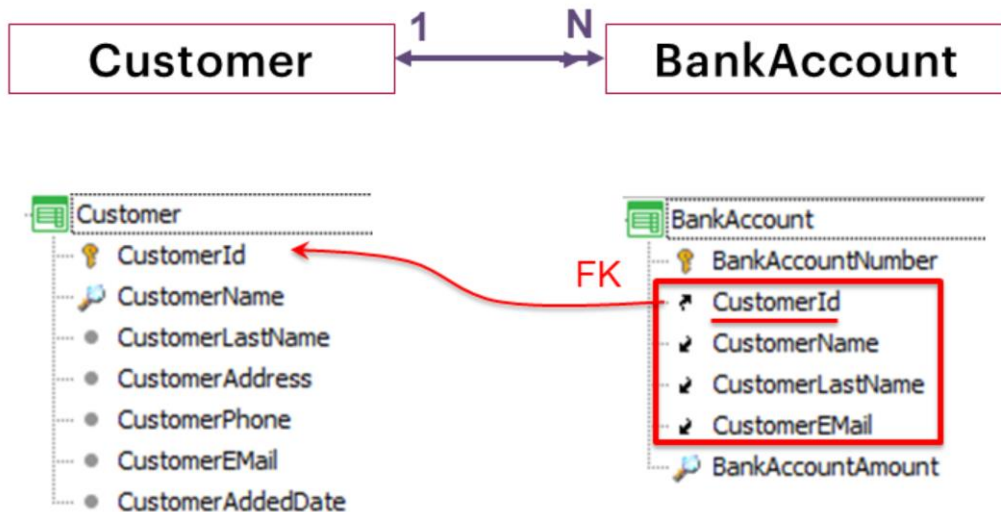
Habíamos visto que mediante transacciones y sus atributos, podemos representar relaciones 1 a N fuertes y débiles entre los actores de nuestra realidad, así como relaciones N a N.



Y habíamos mencionado ya las relaciones 1 a 1 producto de la especialización, tema estudiado en la clase sobre subtipos, como uno de sus casos de uso.

Veremos aquí el otro caso de relaciones 1 a 1, que habíamos introducido antes.

Por ejemplo, cuando la agencia de viajes necesita asociar a cada cliente la cuenta bancaria que se le abre a los efectos de realizar los pagos de los servicios contratados.



Tendremos, además de la transacción Customer, la transacción BankAccount, que se indentifica con un número, y tiene un y solo un cliente asociado.

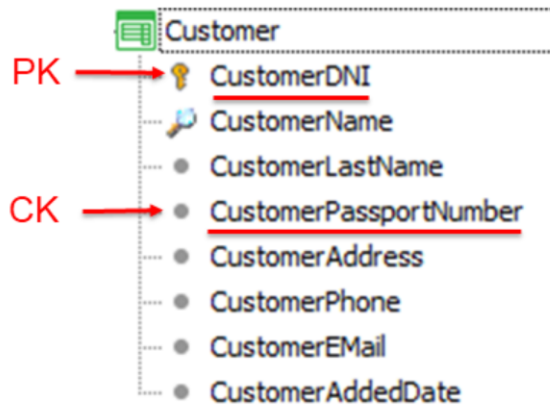
Pero tenemos que hacer algo más porque este diseño corresponde a una relación 1 a N.

Es decir, si bien cada cuenta bancaria podrá tener solamente un cliente, cada cliente podrá estar en muchas cuentas bancarias. Observemos que CustomerId es **clave foránea** en BankAccount, referenciando a la tabla Customer.

Tenemos que conseguir que además ese atributo, CustomerId, no se repita en BankAccount. Dicho de otro modo, que no pueda haber dos o más registros con el mismo valor para ese atributo. Eso es otra forma de decir que queremos que ese atributo sea una **clave candidata**. Volveremos sobre esto luego, cuando veamos los índices.

Clave candidata

- Cliente se puede identificar tanto por DNI como por número de pasaporte. ¿Cuál elegimos como identificador?



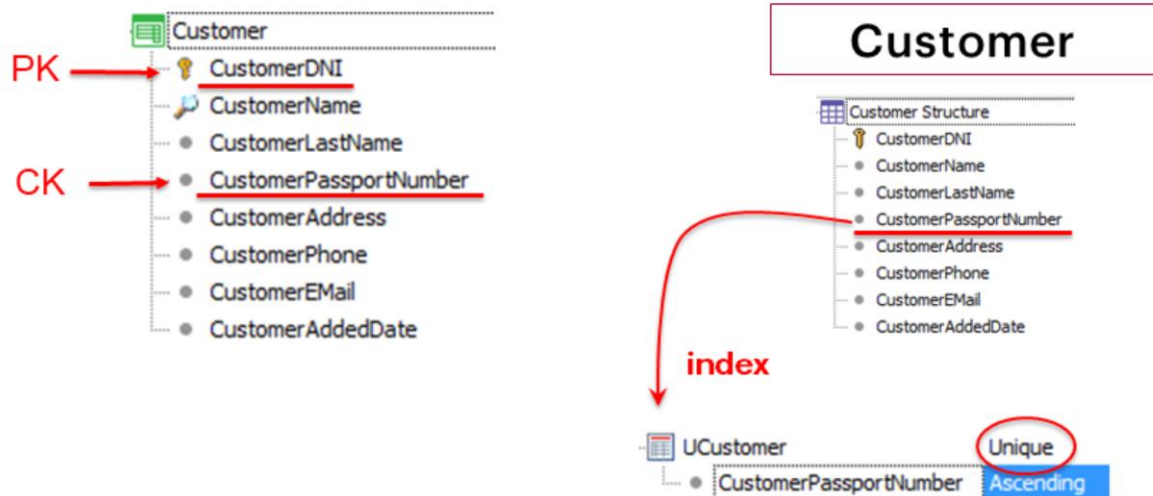
Customer

Para cada nivel de cada transacción es obligatorio definir el atributo o conjunto de atributos que conforman el identificador del nivel. Ese identificador se traducirá a nivel de la tabla física en la clave o llave primaria de la tabla. Con esto estamos diciendo que los valores de este atributo o conjunto de atributos no podrán repetirse.

Pero en muchos casos hay más de un atributo o conjunto de atributos que deben cumplir esa condición. Por ejemplo, para el cliente elegimos identificarlo con su DNI, documento nacional de identidad, expedido por su país, pero también podríamos tener como atributo secundario a su número de pasaporte, que también debe ser único. Como tenemos que elegir a uno de los dos para identificar a la entidad, si no hacemos nada más, el otro quedará como atributo secundario, pudiendo repetirse.

Clave candidata

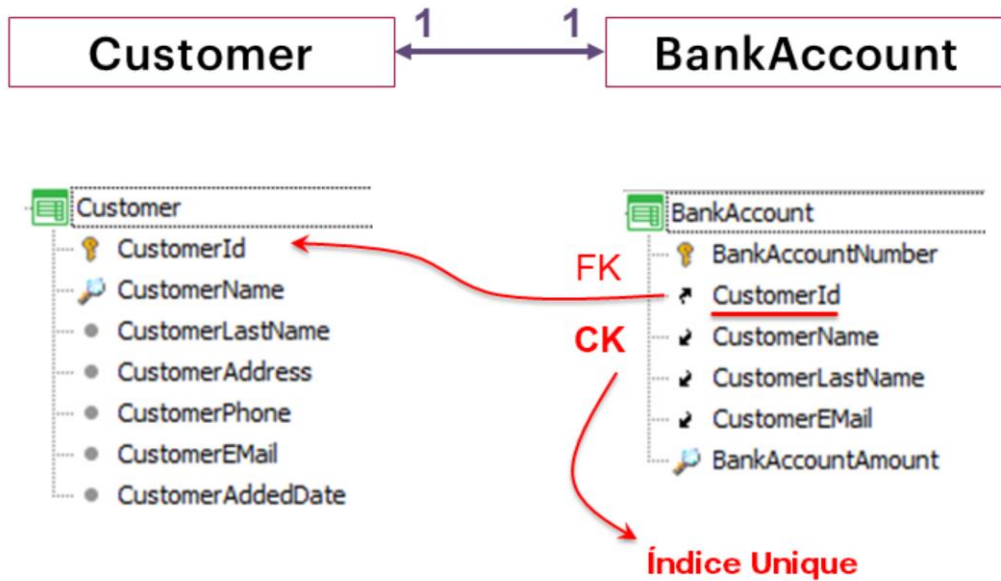
- ¿Cómo especificamos que otro atributo es clave candidata, por lo que debe controlarse su unicidad?



La manera de decirle a GeneXus que el atributo CustomerPassportNumber es una clave candidata es definiendo un índice a nivel de la tabla, sobre ese atributo, e indicando que es un índice que debe controlar que los valores no se repitan: un índice UNIQUE.

Recordemos que los **índices** son vías de acceso eficientes a los datos. Podemos pensar por ejemplo, en un libro de cocina con muchas páginas que contienen recetas, el cual tiene varios índices (índice alfabético, índice por tipos de comidas, etc.). De igual forma, las tablas que almacenan registros, tienen índices también, para ordenar los registros y hacerlos rápidamente encontrables.

Volveremos sobre esto más adelante, donde veremos los índices que son creados automáticamente por GeneXus sobre las tablas y los de usuario. Allí veremos cómo se definen. Por ahora quedémonos con el concepto.



Volveremos sobre esto cuando estudiemos los índices, más adelante...

La solución de diseño que permitirá controlar que el cliente no pueda repetirse en la tabla BankAccount, será definirlo no sólo como clave foránea (lo que se hace automáticamente, tan solo por llamarlo de igual manera que el atributo clave primaria de otra tabla) sino además como clave candidata. Es decir, a través de un índice **Único**.

GeneXus™

The power of doing.

Videos

training.genexus.com

Documentation

wiki.genexus.com

Certifications

training.genexus.com/certifications