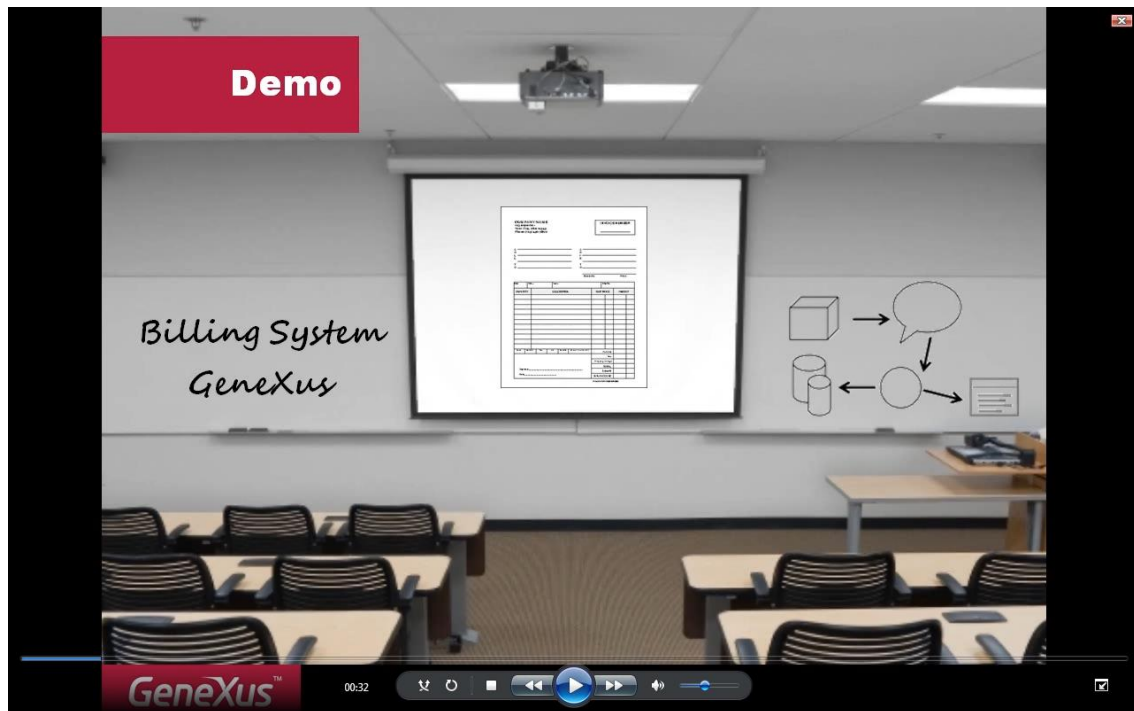


Overview GeneXus - Demo: 1. IDE de GeneXus, creación de base de conocimiento para desarrollar un sistema de facturación y objeto Transacción para ingreso de facturas.



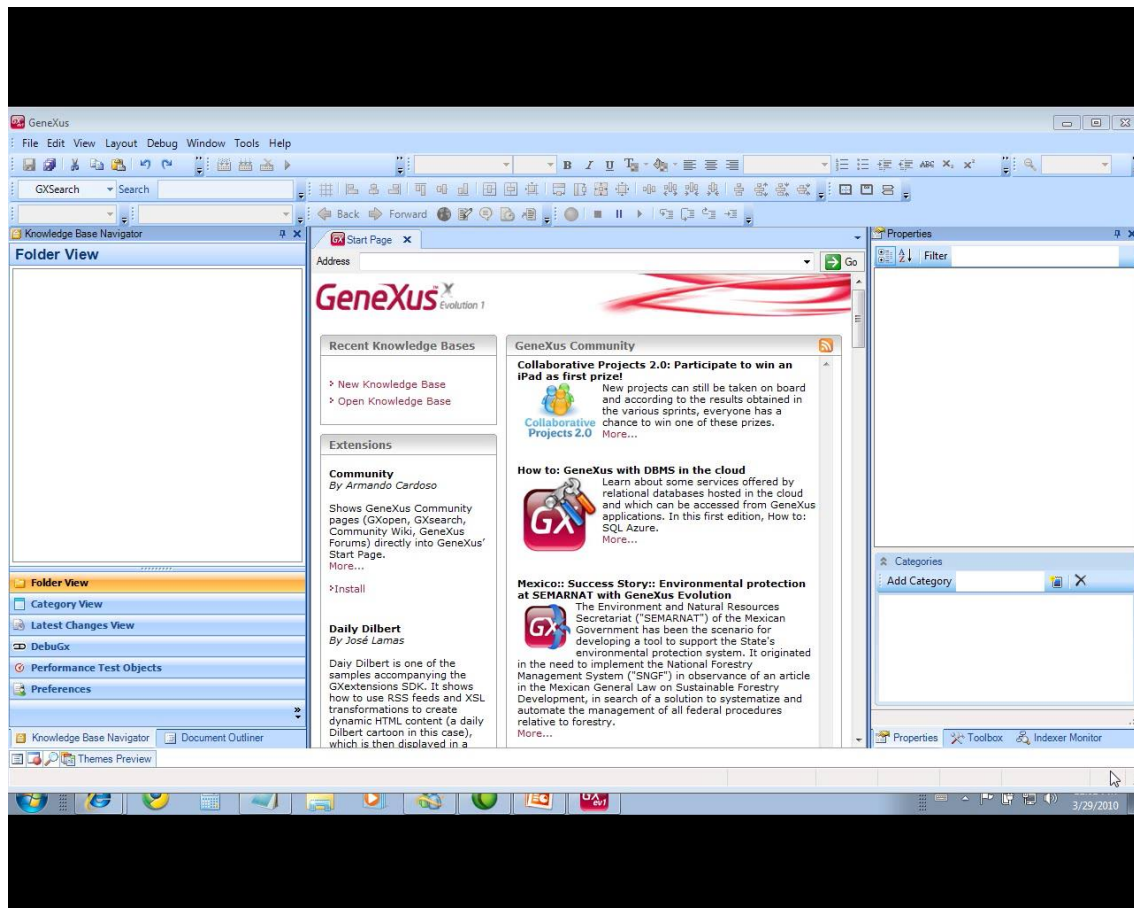
Vamos a ver ahora en una demo las facilidades que brinda Genexus para desarrollar aplicaciones ricas en contenido, usando un entorno de desarrollo amigable y orientado a la necesidad del desarrollador.



Supongamos que se nos pide realizar un sistema informático con tecnología web para automatizar las tareas de una empresa de venta de productos. Concretamente se necesita un sistema de facturación.

Lo que debemos hacer entonces es describir la realidad de la empresa utilizando objetos Genexus. A partir de ese **conocimiento** Genexus automáticamente **creará** la base de datos y **generará** los programas de la aplicación por nosotros.

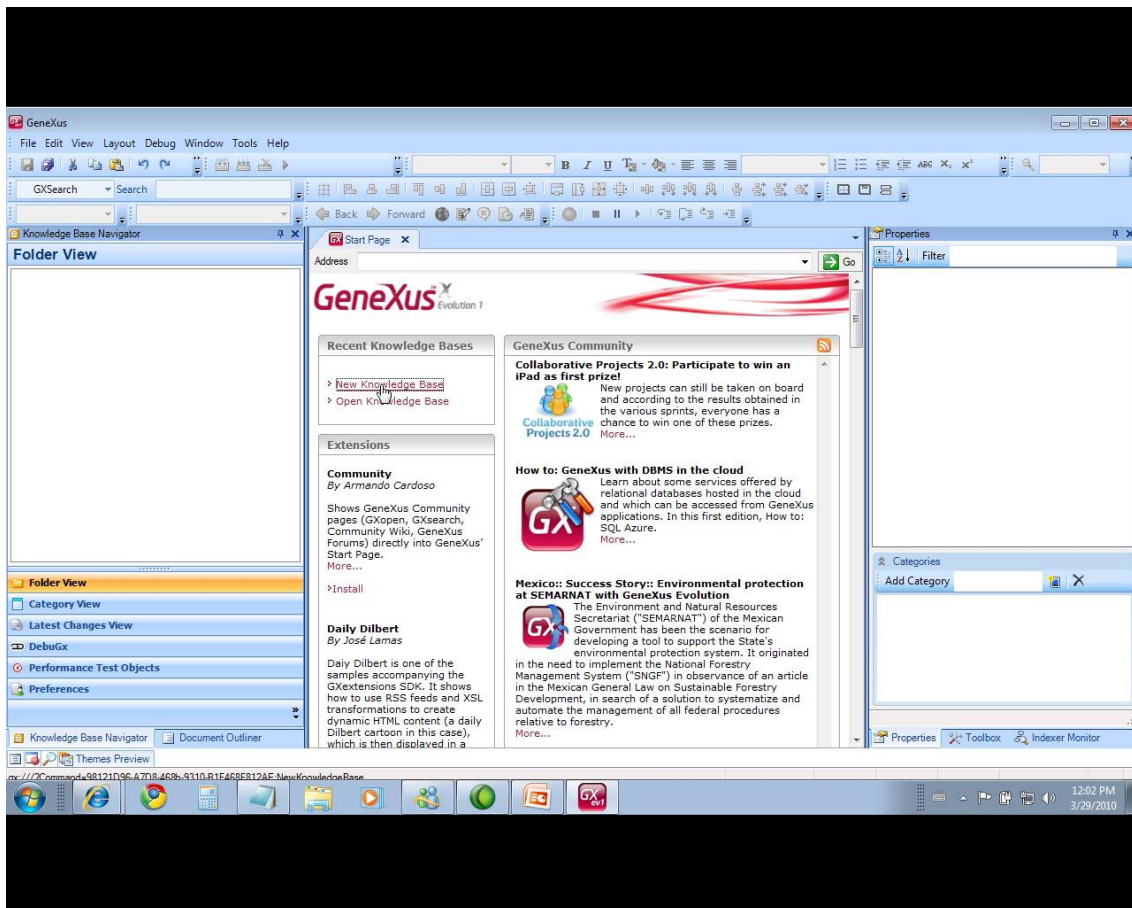
Para eso abrimos el ambiente de desarrollo de Genexus.



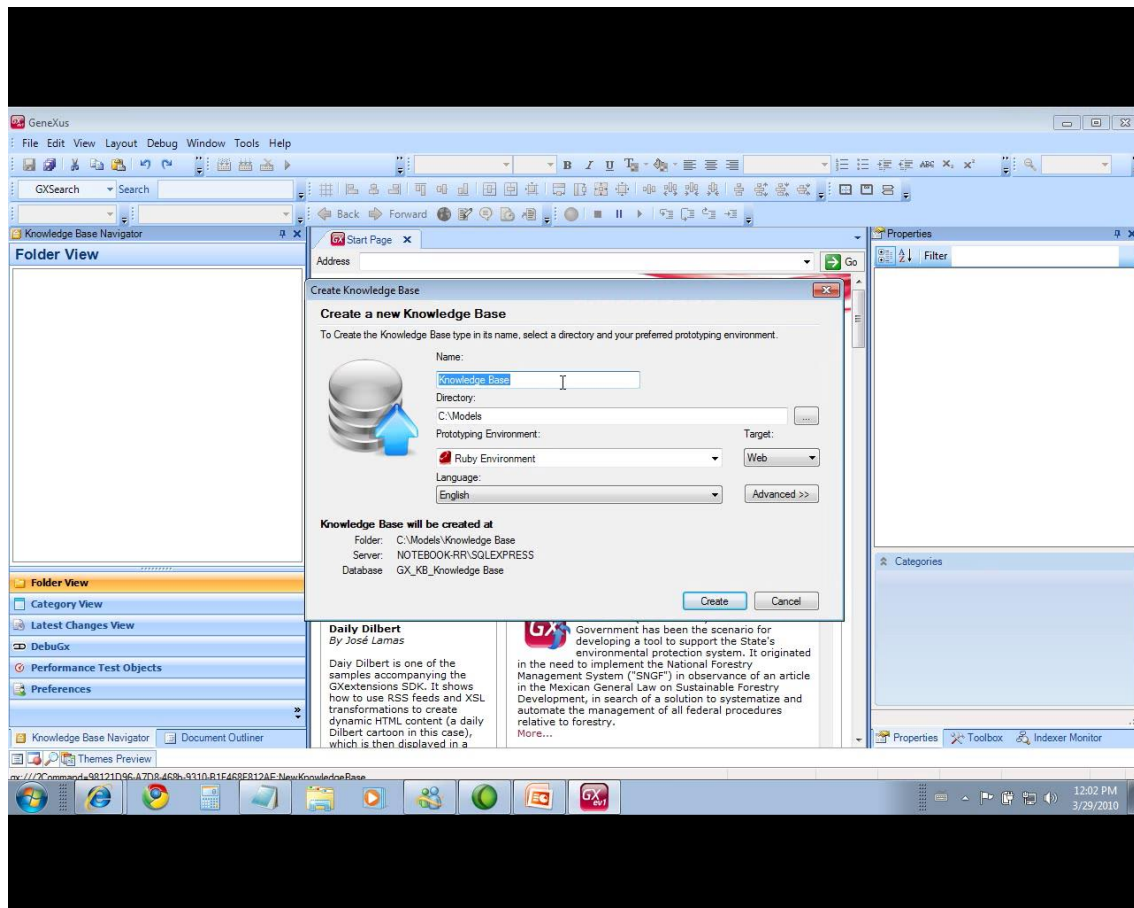
Aquí tenemos el entorno de desarrollo integrado de Genexus. Es un IDE moderno con una apariencia la cual están acostumbrados los desarrolladores, que cuenta con una arquitectura **extensible** que permite agregar componentes y aumentar la capacidad del Genexus inicialmente instalado.

Además, una cosa **interesante** es que dentro del mismo entorno de desarrollo, el analista esta conectado con la comunidad **Genexus** y puede ver noticias, actualizaciones, soluciones publicadas por otros desarrolladores como él, etc.

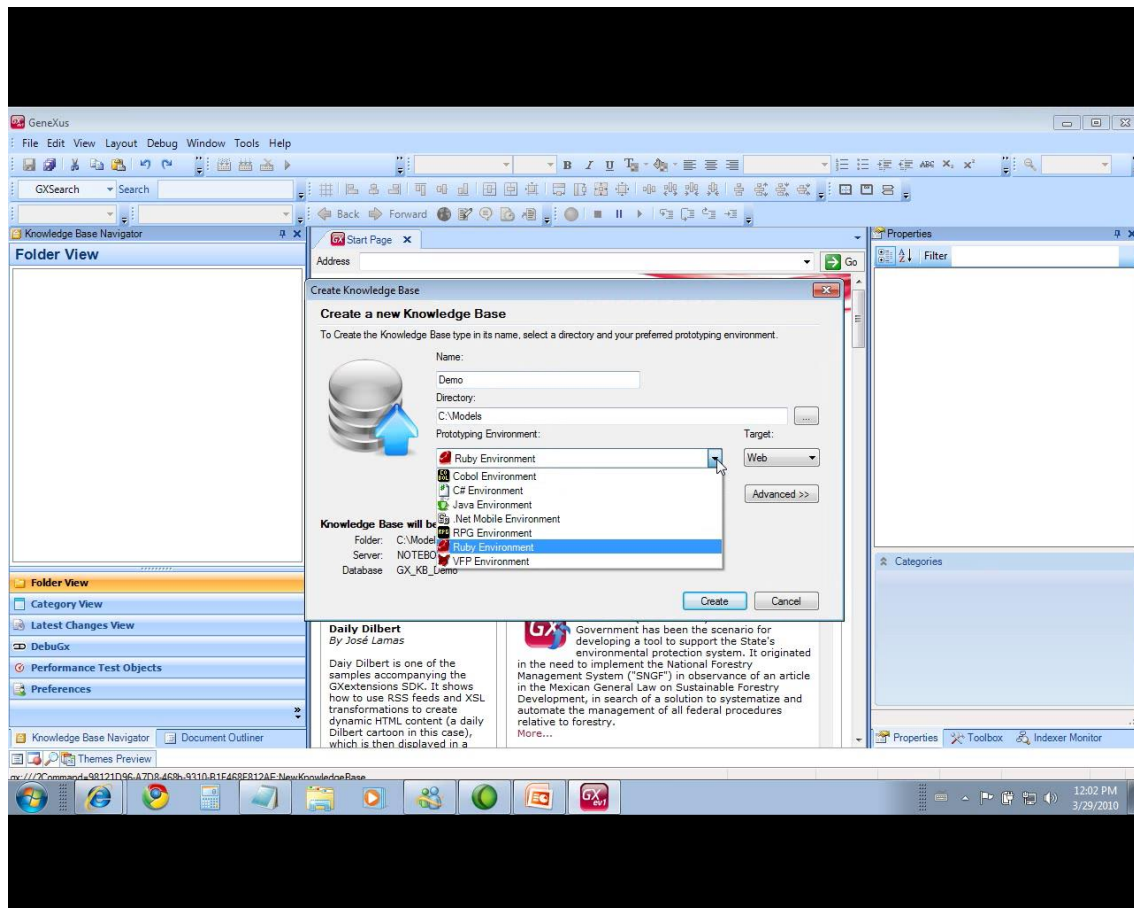
Para empezar a trabajar con **Genexus** lo primero que tenemos que hacer es crear una base de conocimiento.



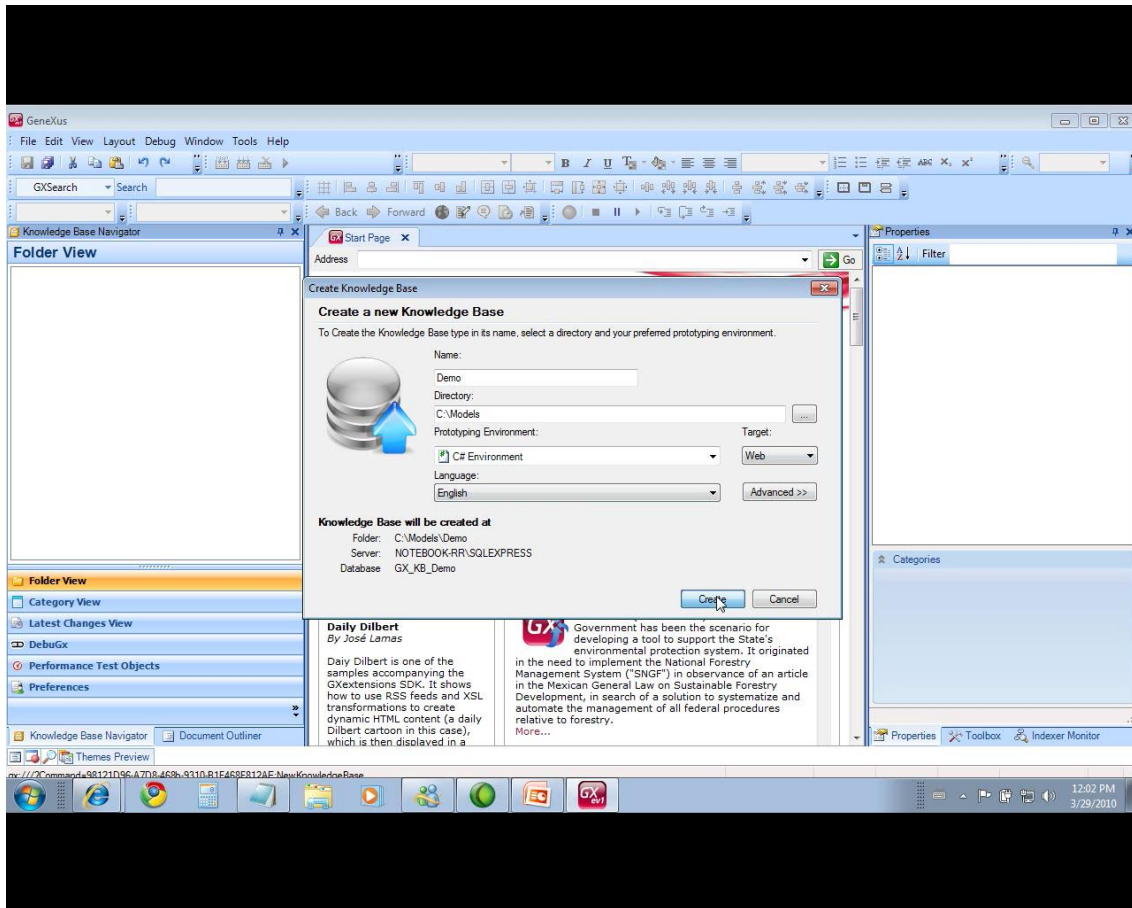
Para eso, hacemos clic en **New Knowledge Base** y



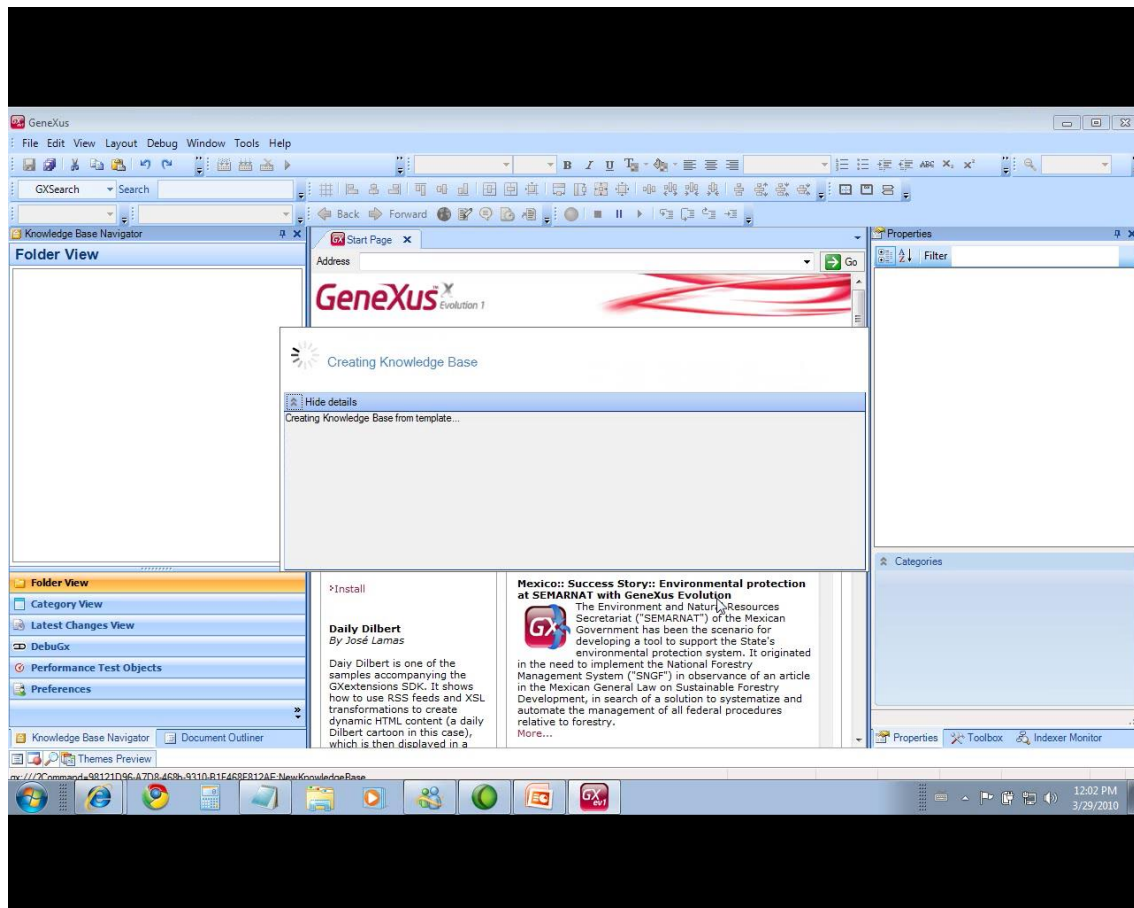
en este cuadro de diálogo ponemos el nombre de la base de **conocimiento**, la carpeta directorio donde vamos a almacenar dicha base de **conocimiento**,



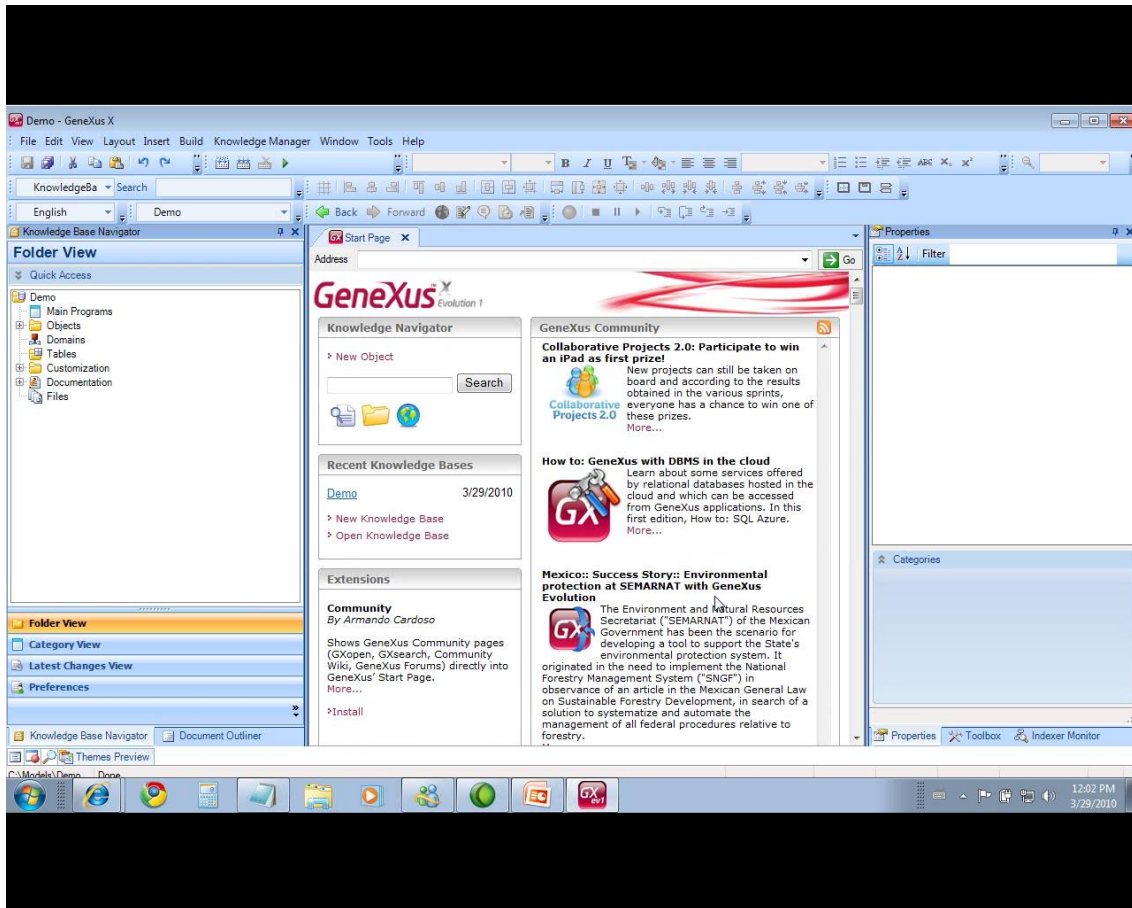
el **lenguaje** que vamos a utilizar para generar la aplicación,



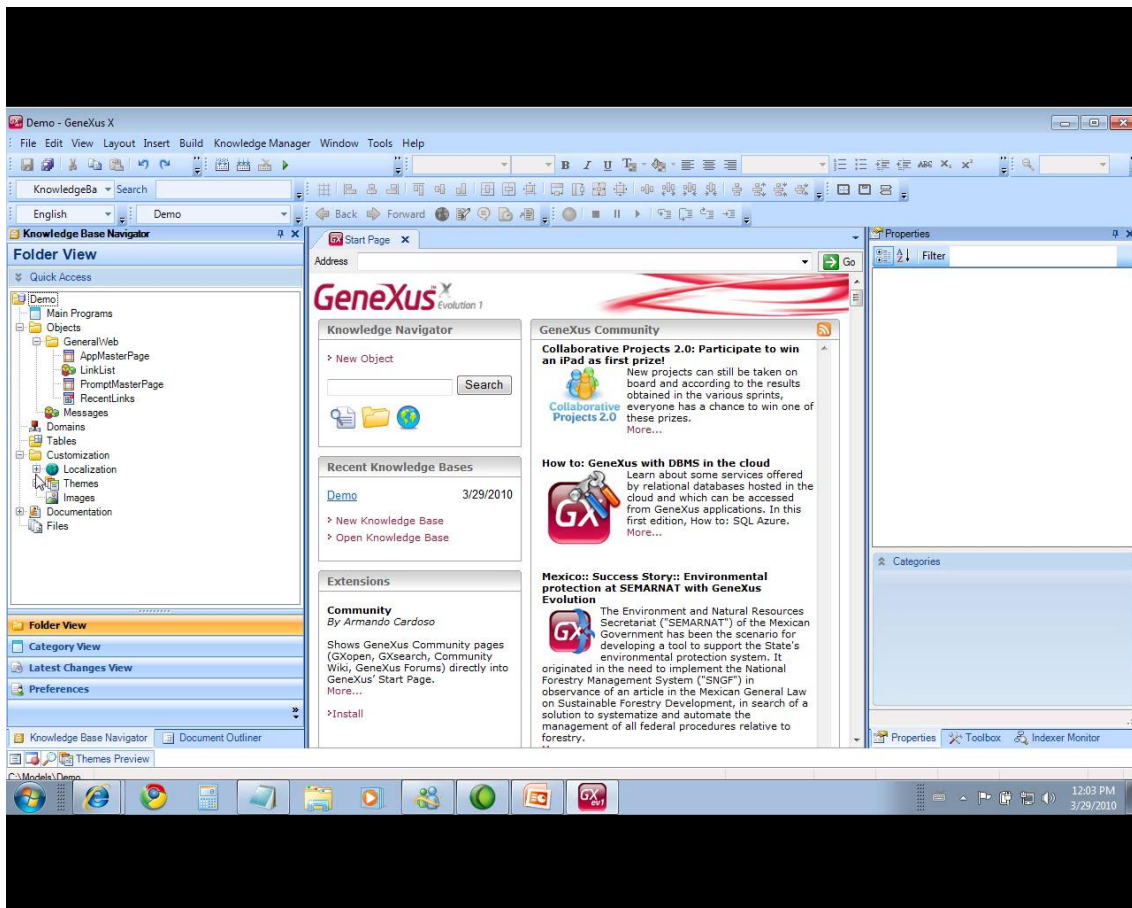
dejamos interface web, idioma inglés y presionamos el botón de create.



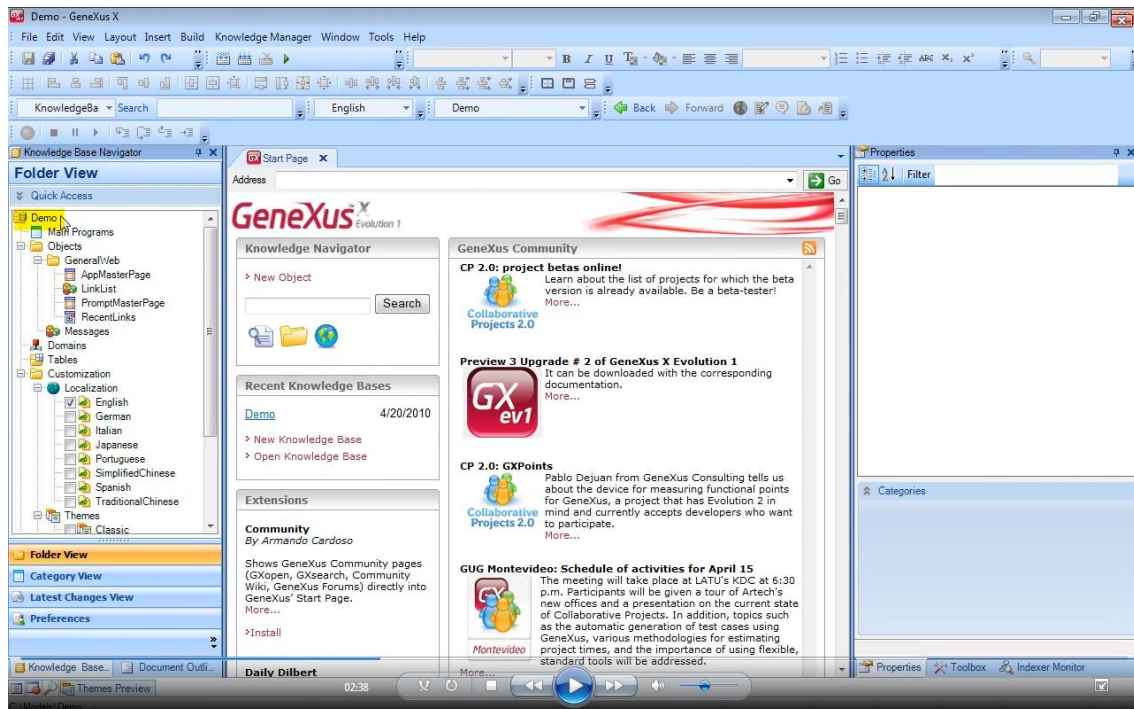
En este momento se están creando todos los objetos y estructuras necesarias para que la base de conocimiento quede lista para empezar a trabajar. **Genexus** configura estos objetos inicializándolos con valores por **defecto** que luego podremos **cambiar** y **personalizar**, tanto el comportamiento como la apariencia de nuestra aplicación.



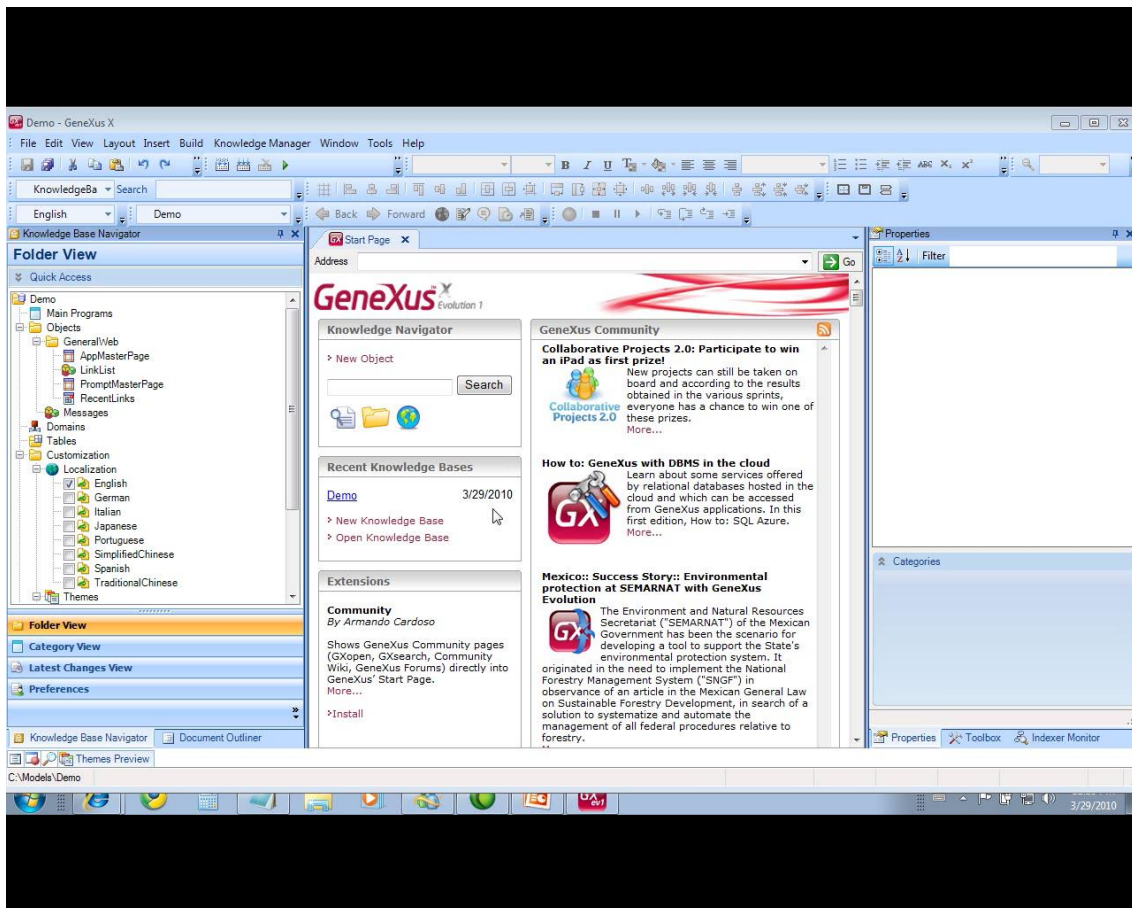
Aquí tenemos creada entonces nuestra base de conocimiento. Observemos que en el Knowledge Base Navigator



tenemos varios **objetos** que Genexus creó.



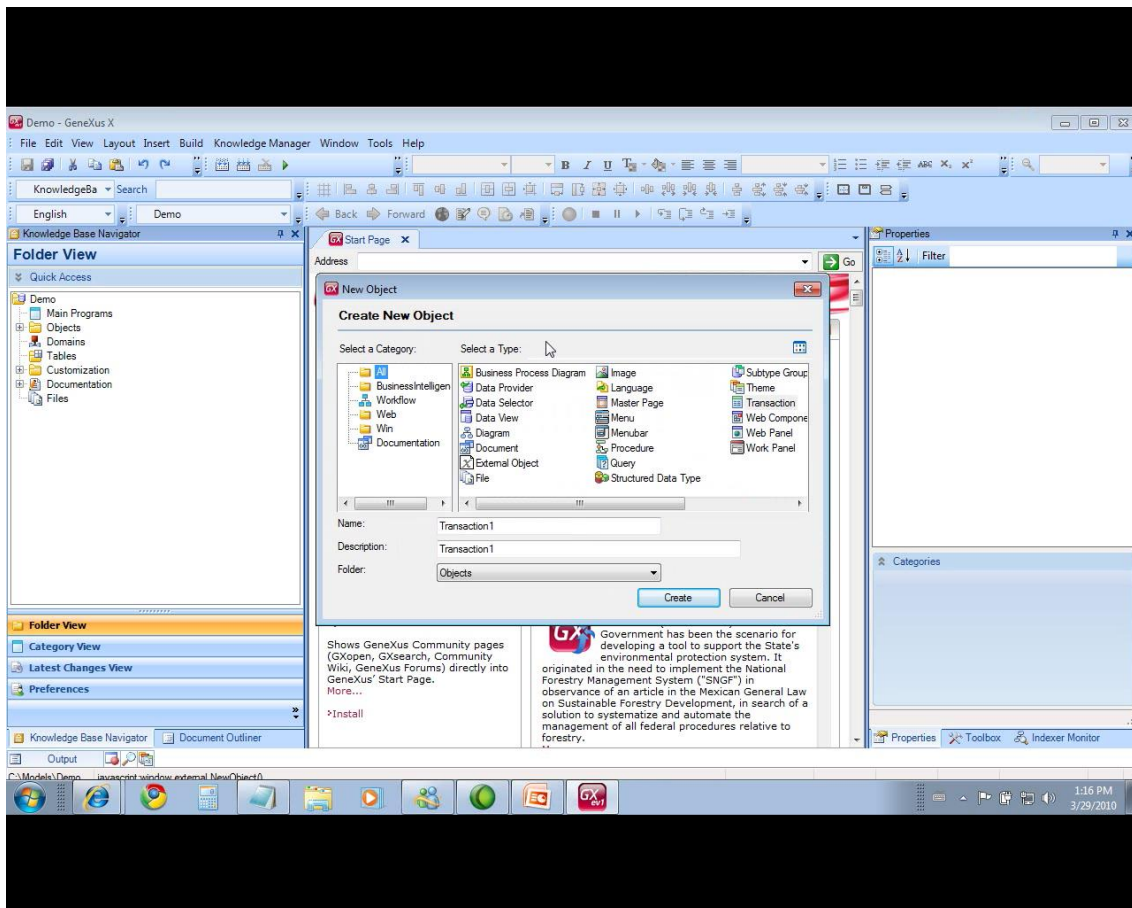
Observemos que tenemos varias referencias a la **Knowledge Base** o KB en la que estamos trabajando, aquí por ejemplo en el Knowledge Base Navigator,



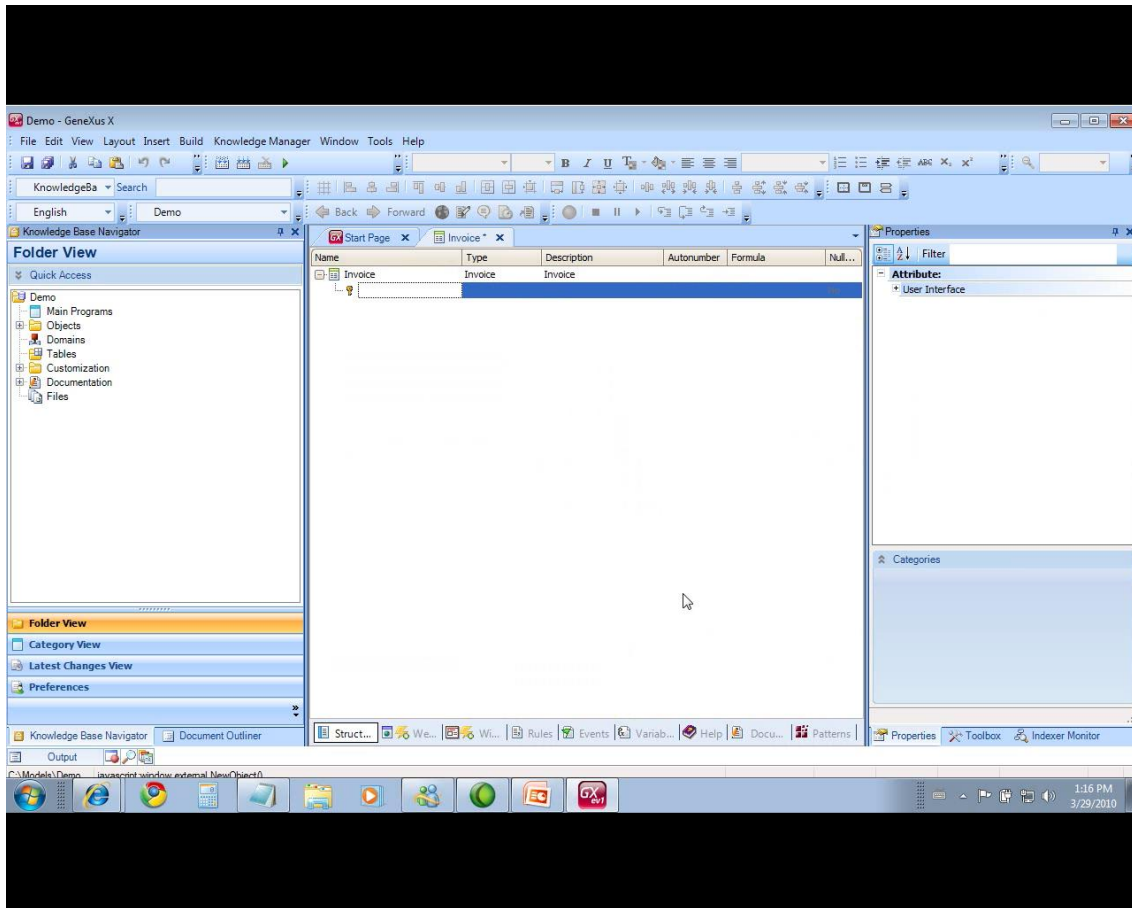
aquí con el nombre y la fecha de creación y también en la barra de estado donde se indica el **path** completo donde se creó la KB.

Se nos encomendó la tarea de crear un sistema de facturación, o sea que vamos a tener que ingresar facturas. El objeto Genexus que se **utiliza** para **ingresar y editar** información a la base de datos en forma **interactiva** es el objeto transacción.

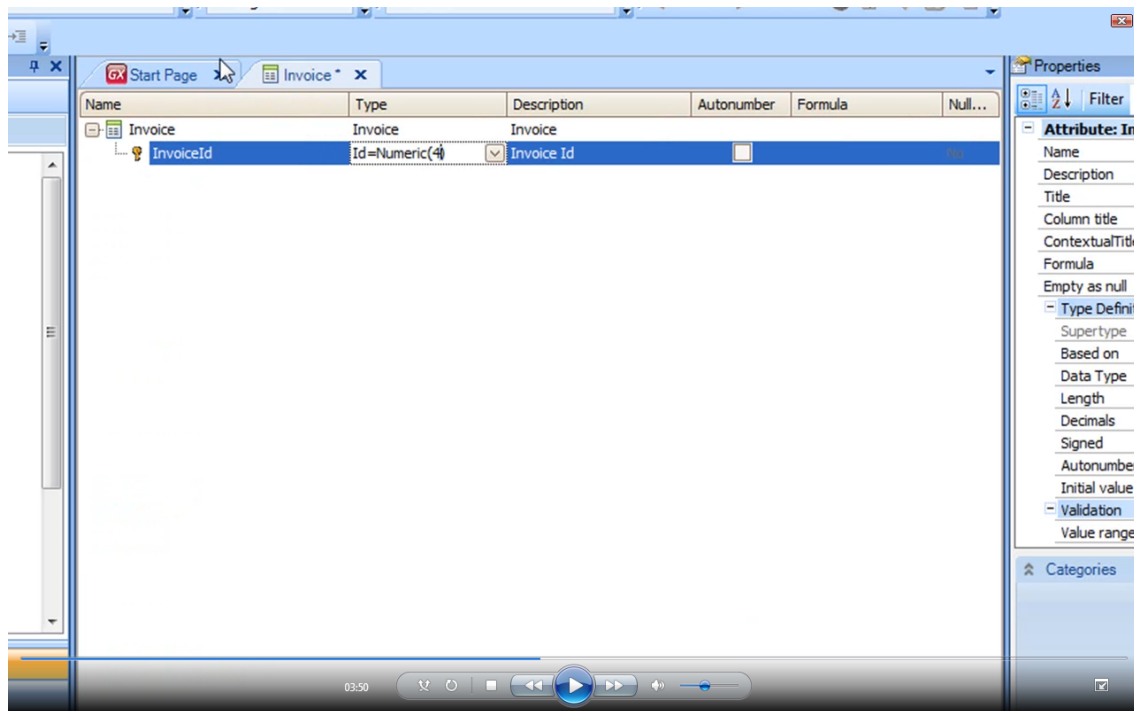
Así que vamos a tener que crear una transacción de facturas. Hagamos clic en el link de New Object y



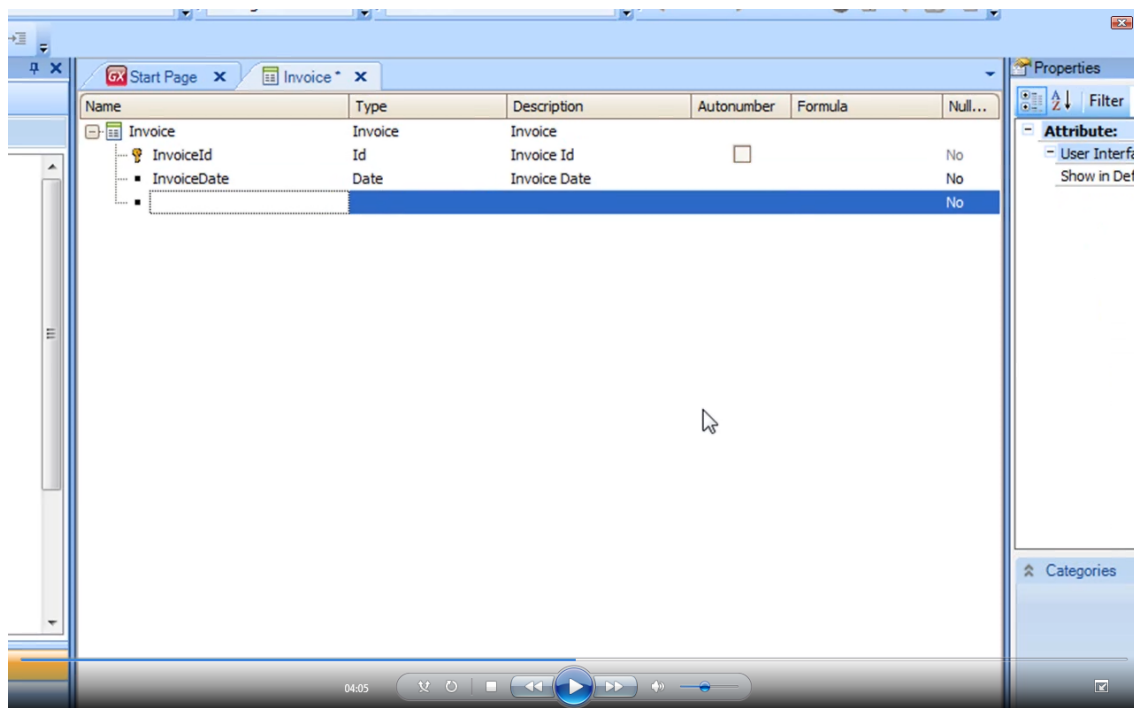
en el cuadro de dialogo de crear un objeto nuevo elegimos un objeto de tipo **transacción**, le ponemos el nombre, Invoice en este caso y



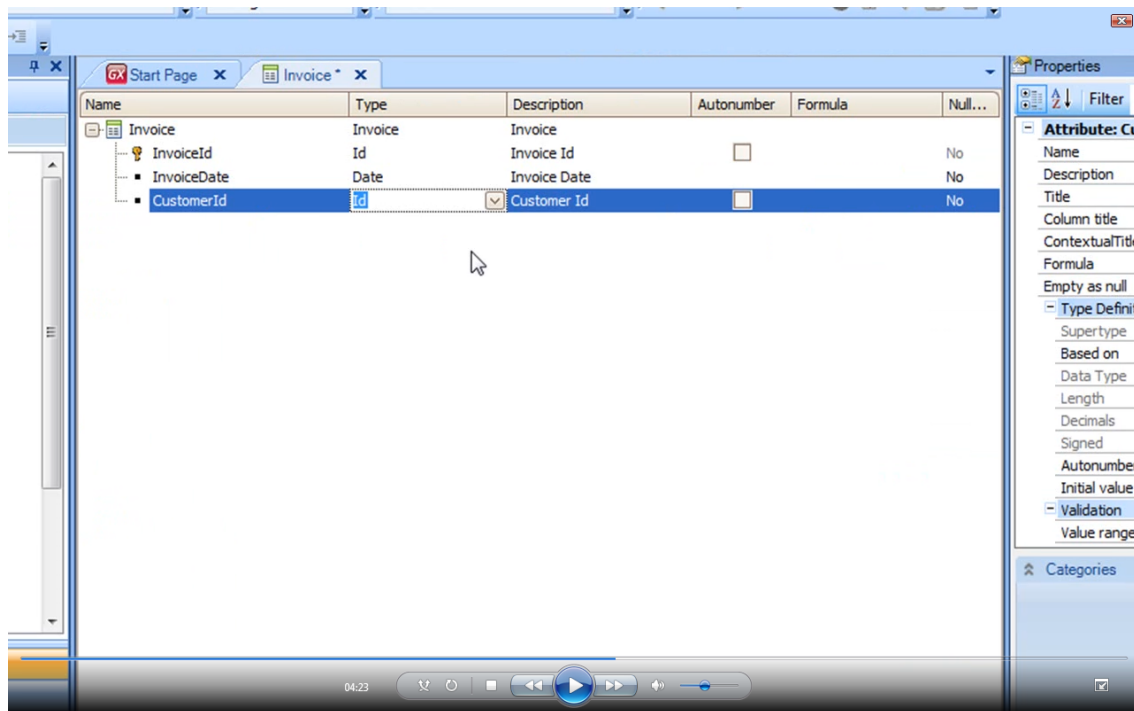
vamos a ingresar ahora los atributos que caracterizan a una factura. Una factura tiene **por ejemplo** un número que la identifica en forma única. Digamos un InvoiceId,



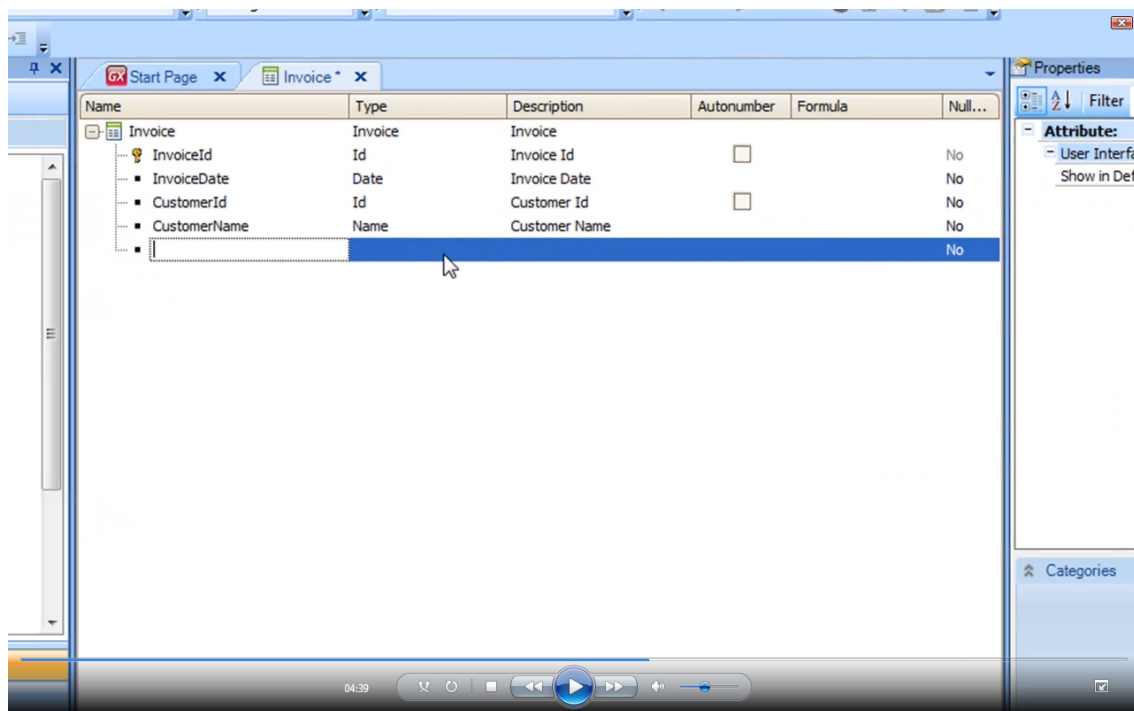
del tipo numérico, pero en este caso me voy a crear un tipo de datos Id del tipo numérico porque ya preveo que vaya a tener varios identificadores que sean del tipo Id y vamos a querer que este número sea auto numerado.



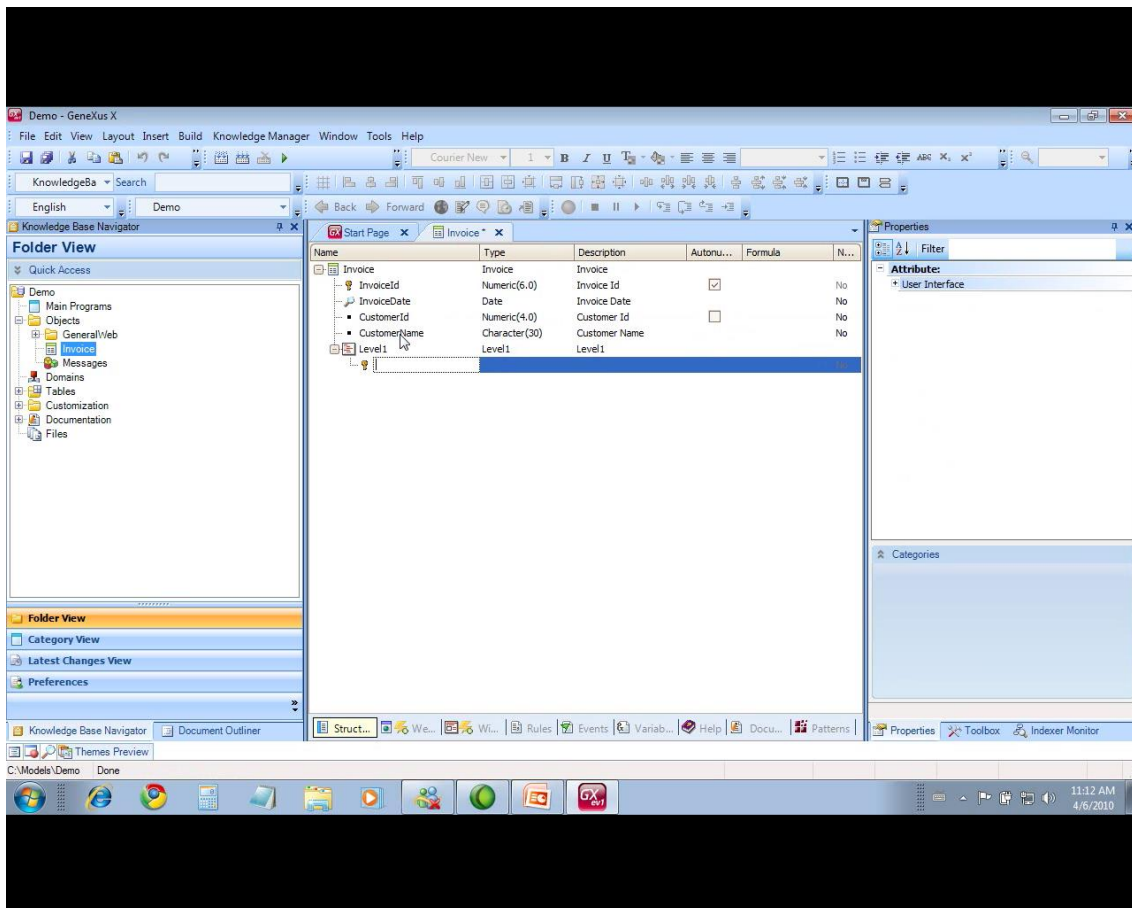
Una factura tiene también una fecha InvoiceDate. Observemos que en este caso GeneXus asignó **automáticamente** el tipo de datos Date debido al nombre que le puse yo al atributo.



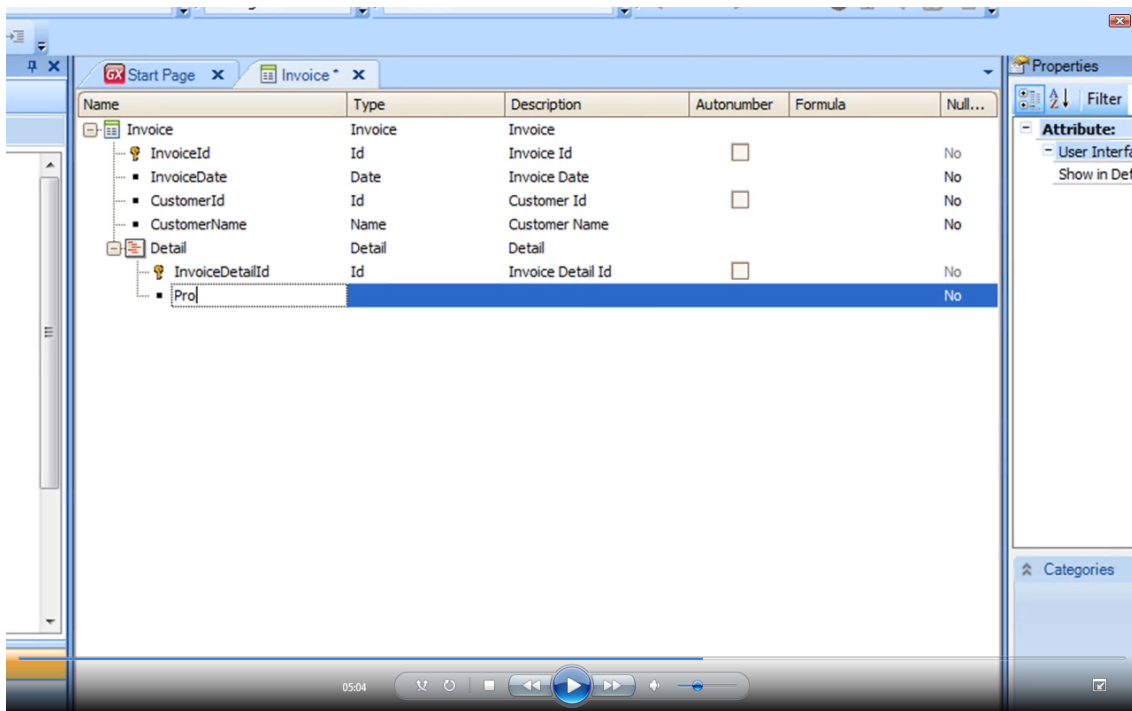
Una factura también tiene un **cliente**. Y aquí otra vez, GeneXus **asignó** el tipo de datos **Id**, que es un dominio, un tipo de datos nuevo, al atributo CustomerId debido a una asociación en el nombre. Esto es conocimiento que quedó almacenado en la base de conocimientos y **GeneXus** lo está utilizando.



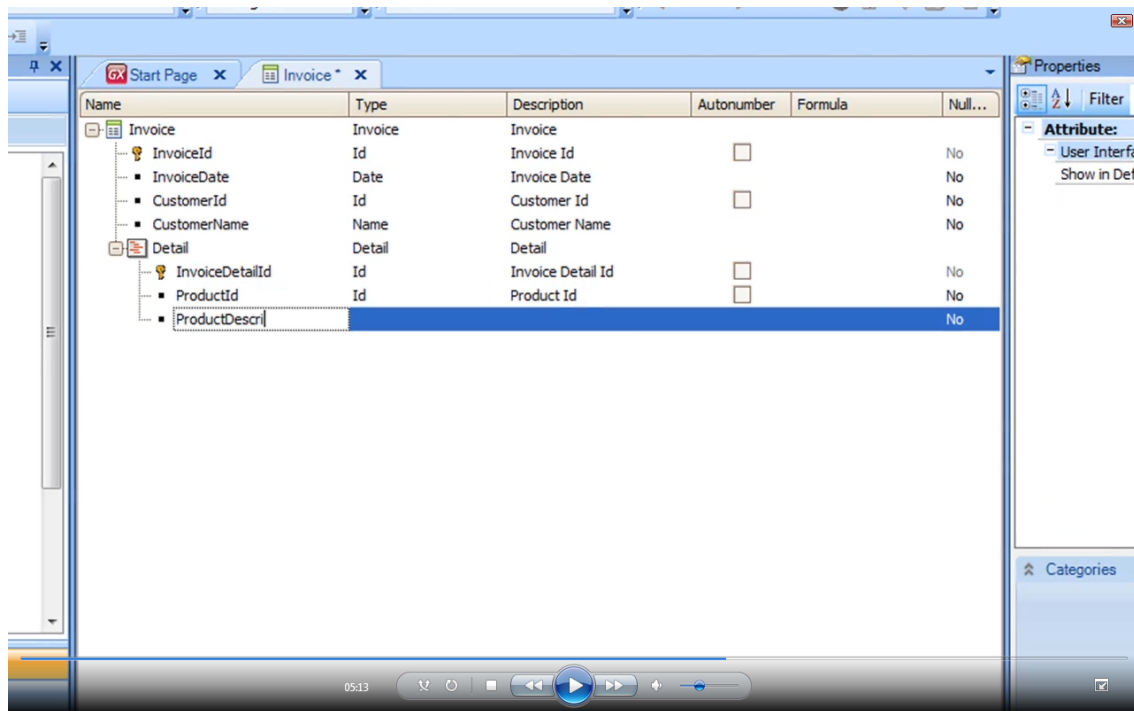
Vamos a crear el nombre de cliente. Aquí voy a crear un tipo de datos Name. Character de 30. Y hasta aquí hemos creado la parte fija de la factura, es decir el **cabecal** de la factura.



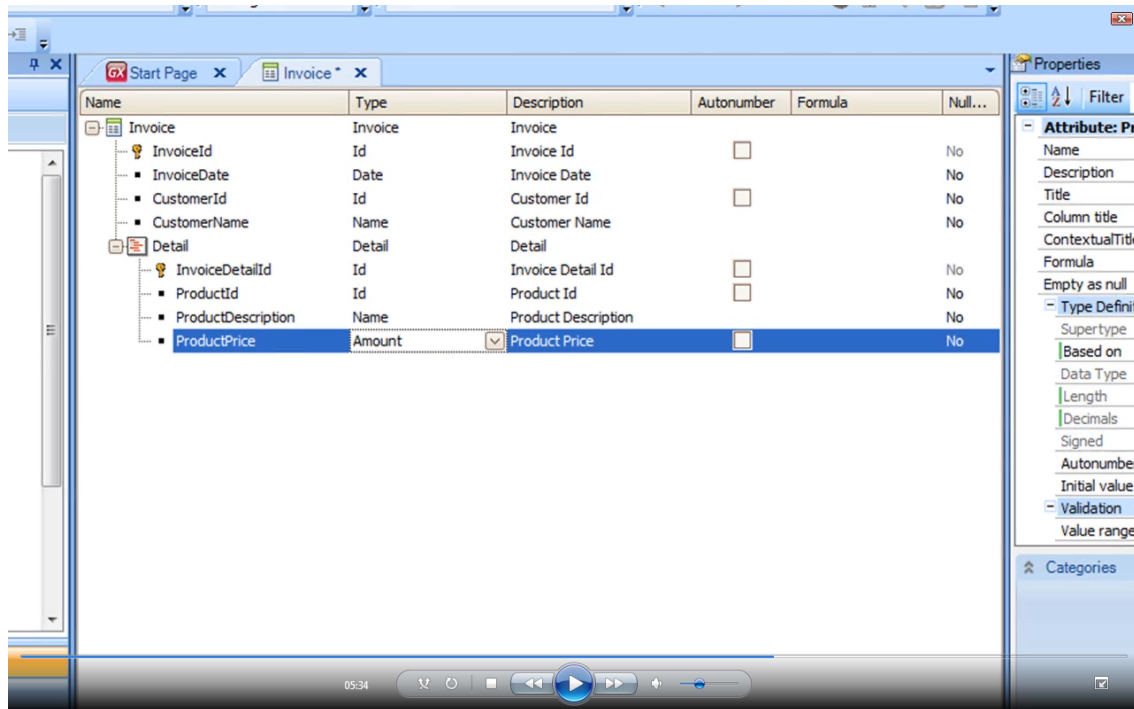
Vamos a ingresar ahora la línea de **detalle** que contiene los productos a ser facturados. Agregamos un nivel **subordinado** y a cada línea le damos un identificador de línea.



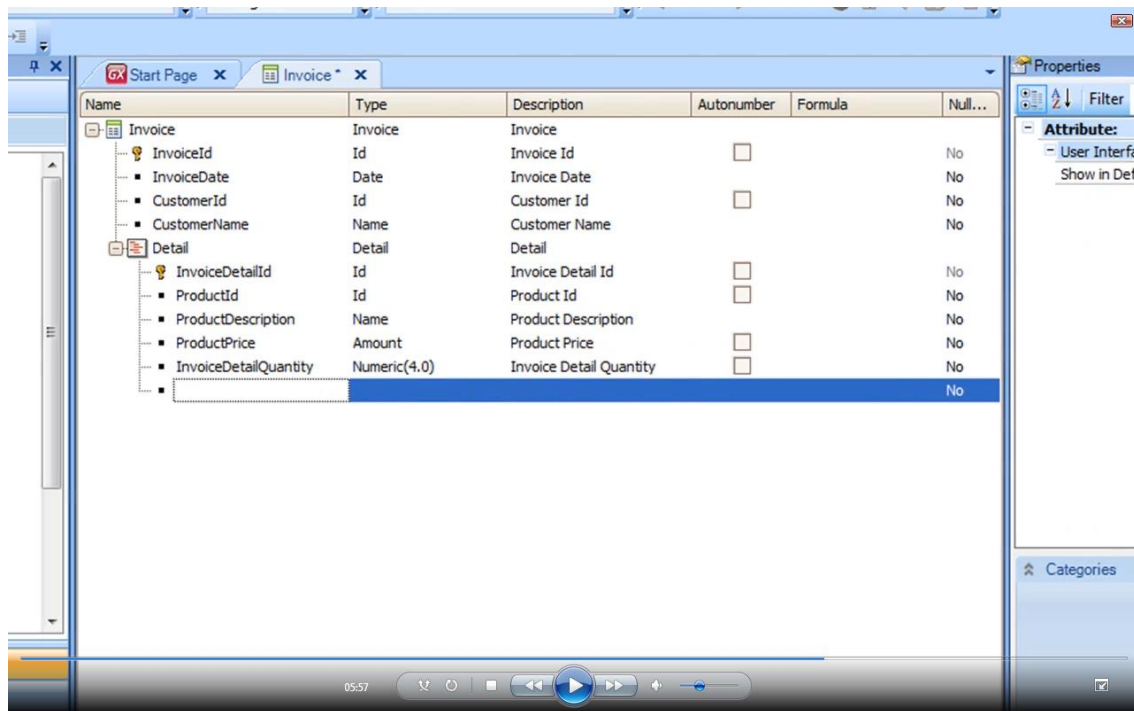
Una línea tiene también un producto.



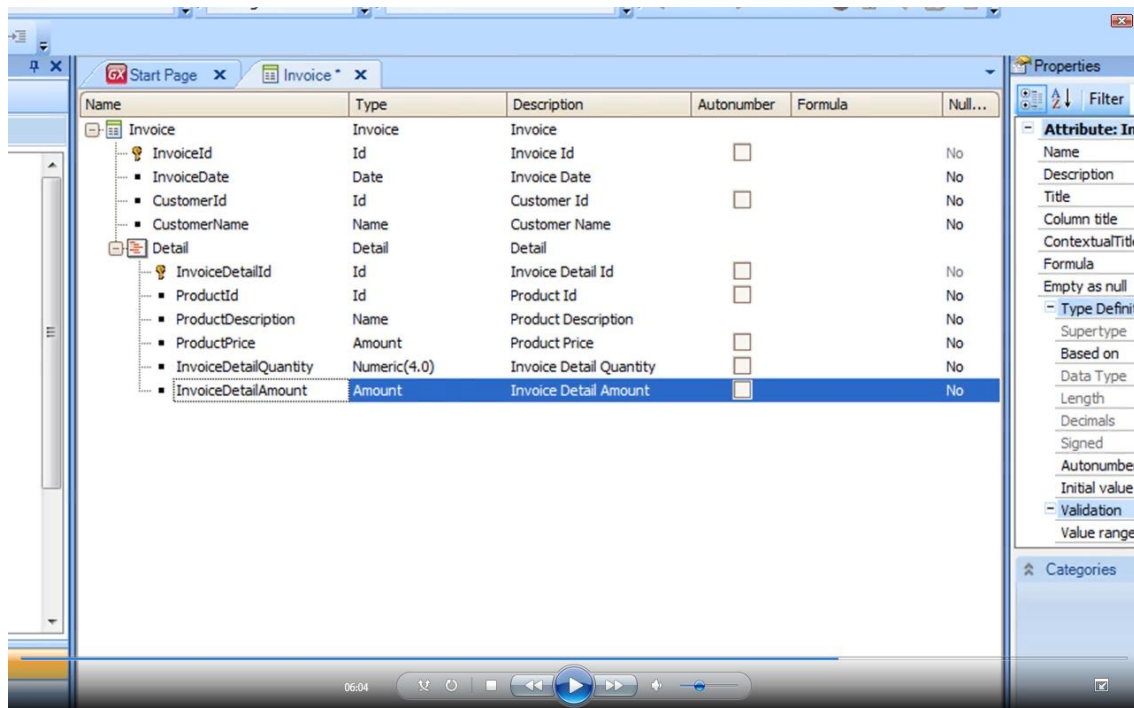
El producto tiene una **descripción**



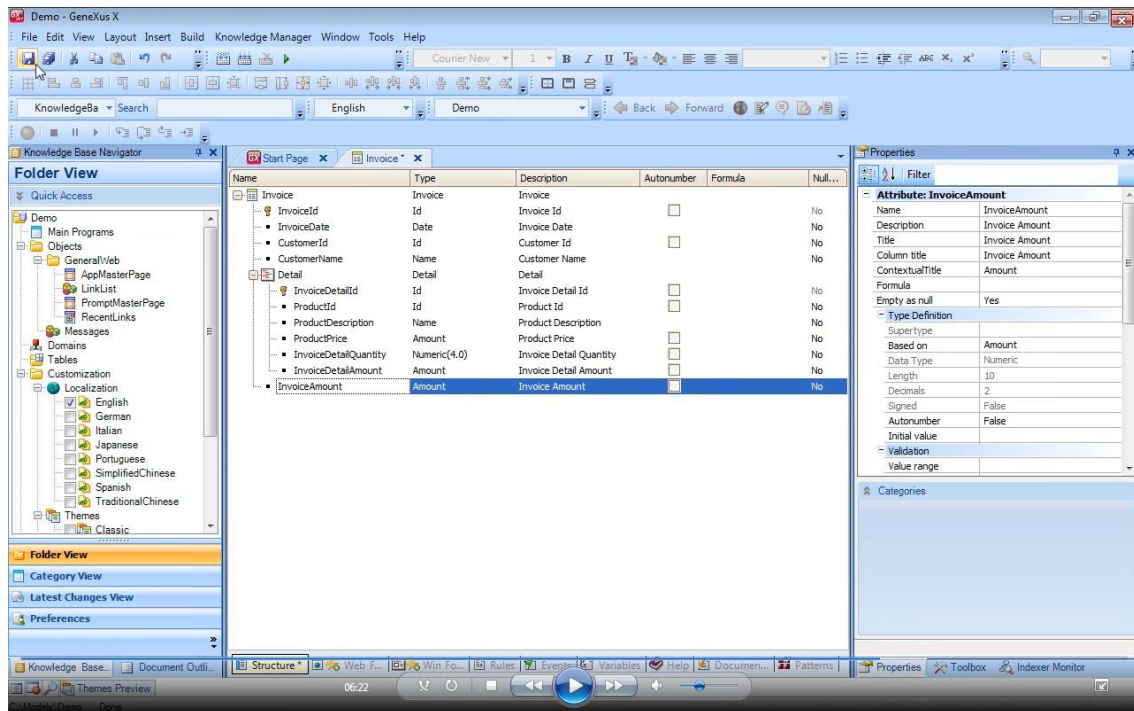
y un precio. Y aquí voy a crear un tipo de datos **nuevo**, numérico de 10.2. Estos tipos de datos **nuevos** llamado dominios nos dan la ventaja de que todos los atributos que basemos en un dominio, luego si cambiamos ese dominio cambia automáticamente todos los atributos basados en él. Una línea tiene entonces también



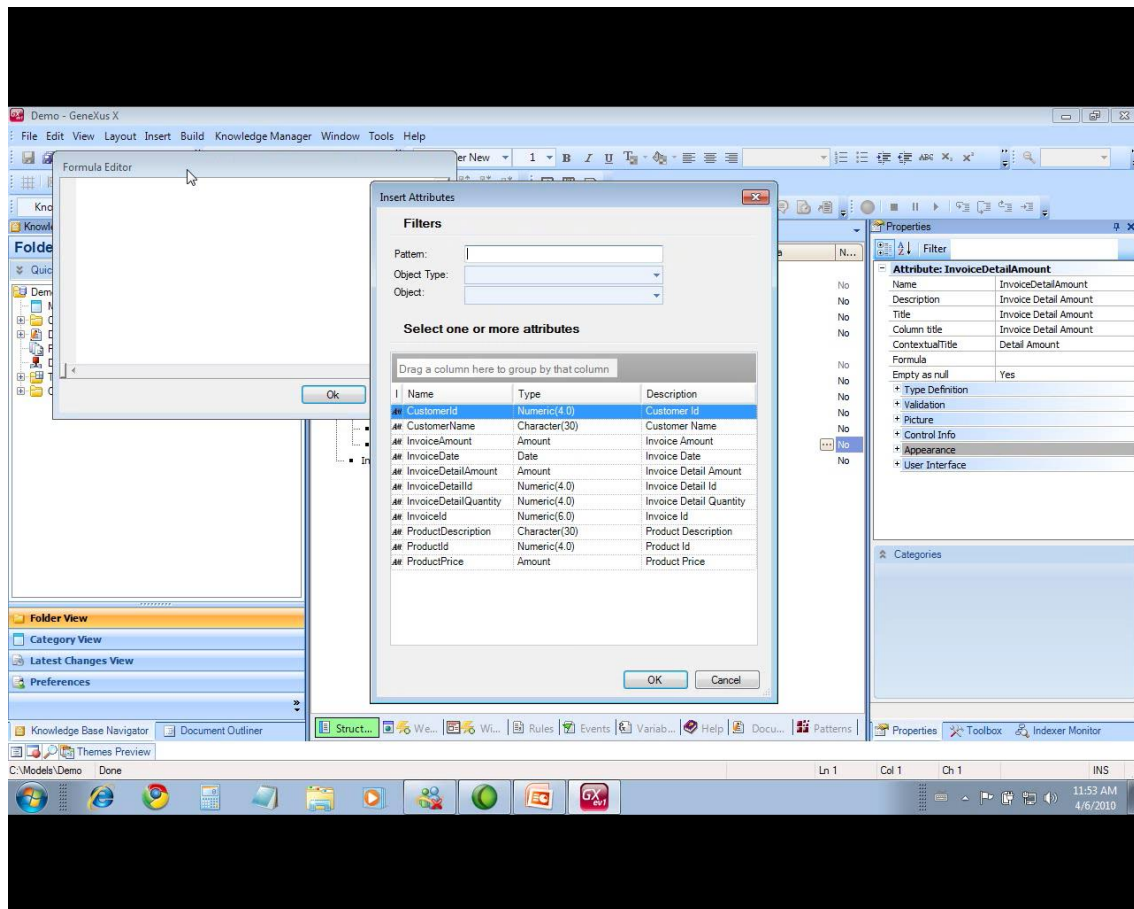
la cantidad que vamos a facturar



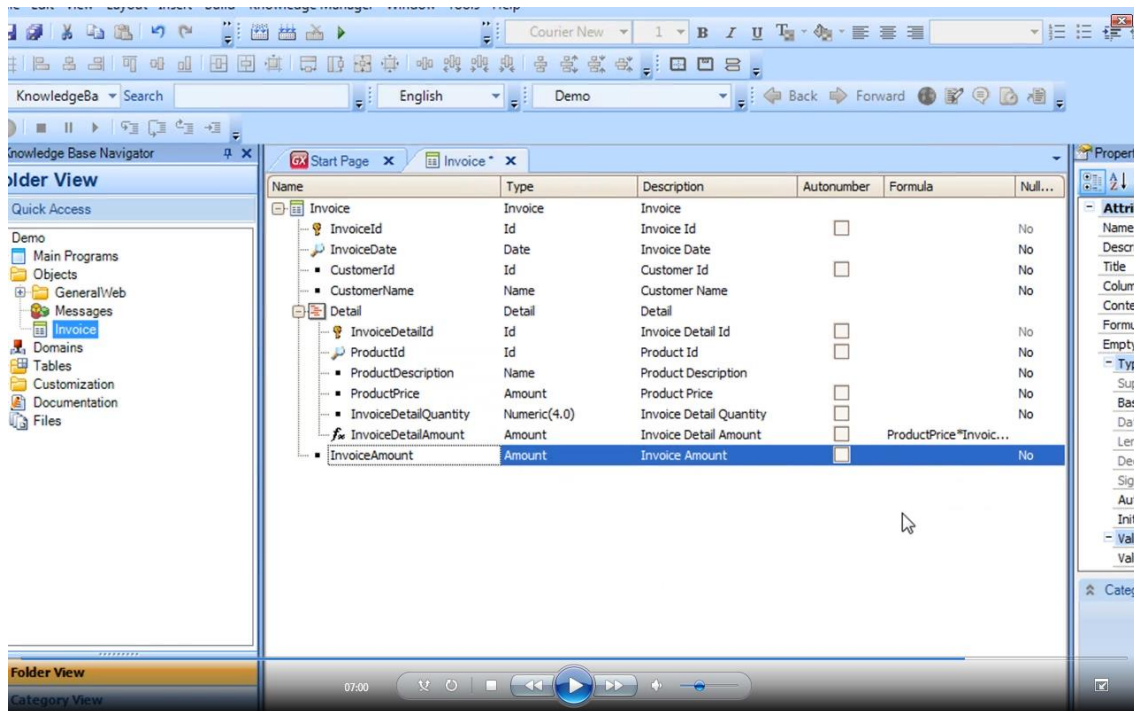
y el importe de la línea. Observamos que como el atributo terminaba en Amount, Genexus automáticamente encontró que podía ser del tipo de datos del dominio que habíamos definido. Para terminar agregamos el total de la factura.



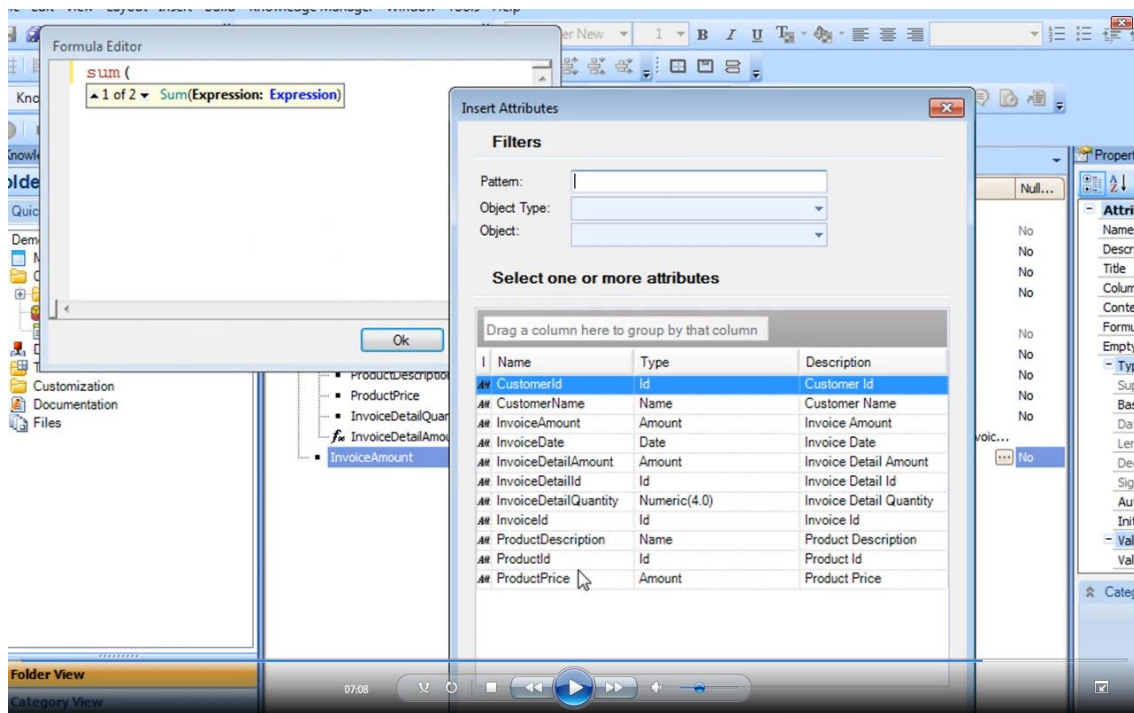
Salvamos, ahora bien, el importe de la **línea**, en realidad, podríamos calcularlo como el precio del producto por la cantidad de producto facturado. Así que para el importe de la línea vamos a calcularlo como una fórmula.



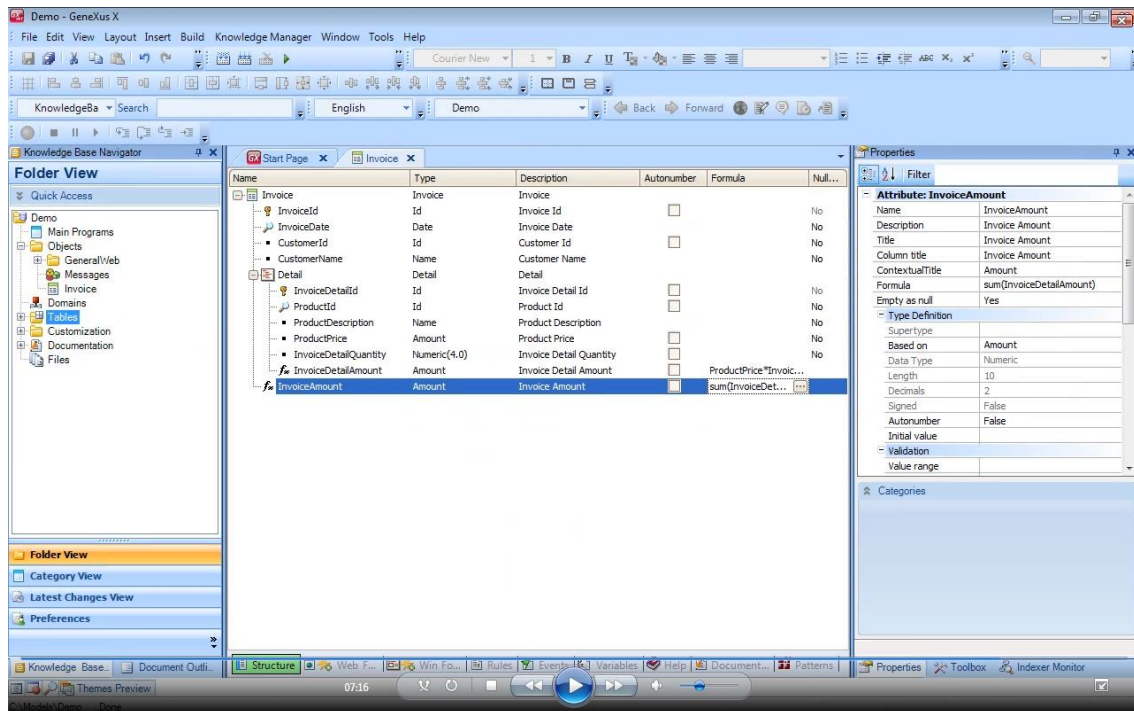
Y definimos la fórmula como el producto del precio del producto por la cantidad facturada.



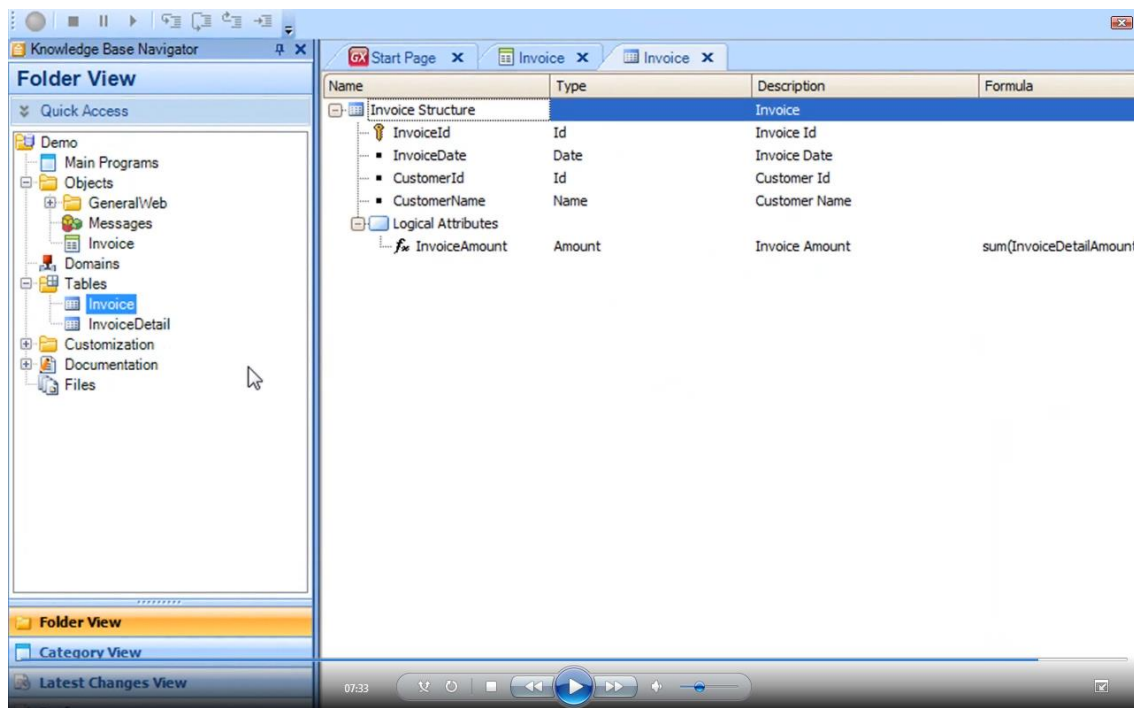
También la **suma** de todas las líneas es el total de la factura.



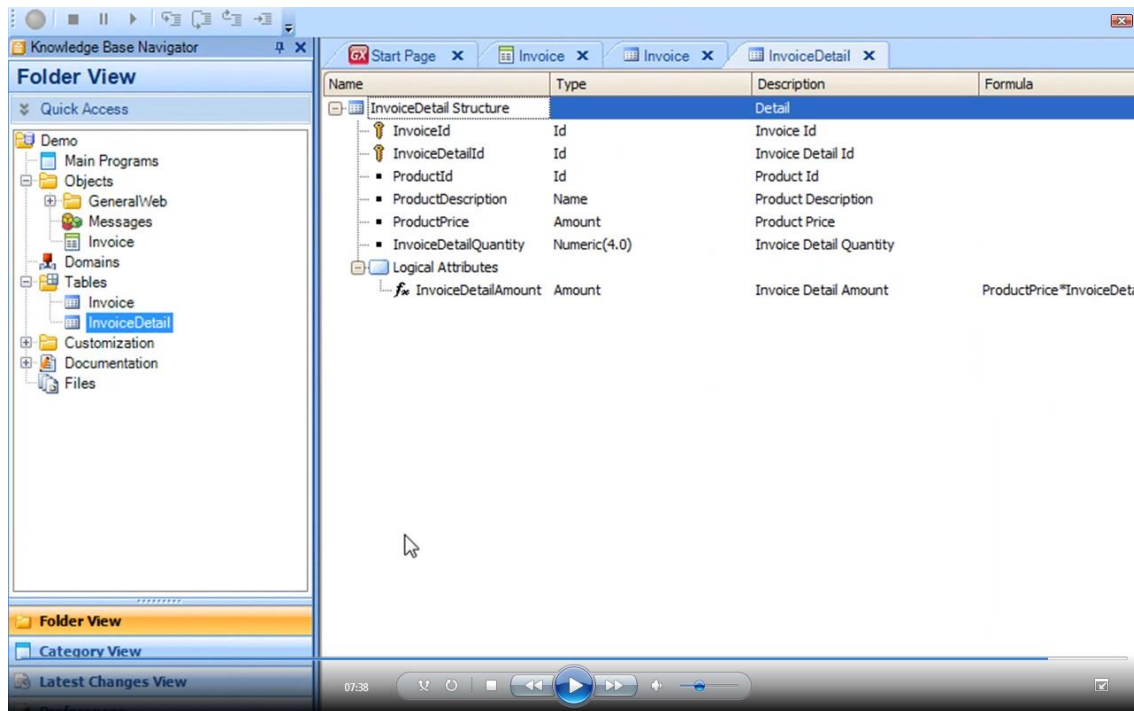
Así que vamos a calcular el atributo **InvoiceAmount** como la suma de las líneas de la factura.



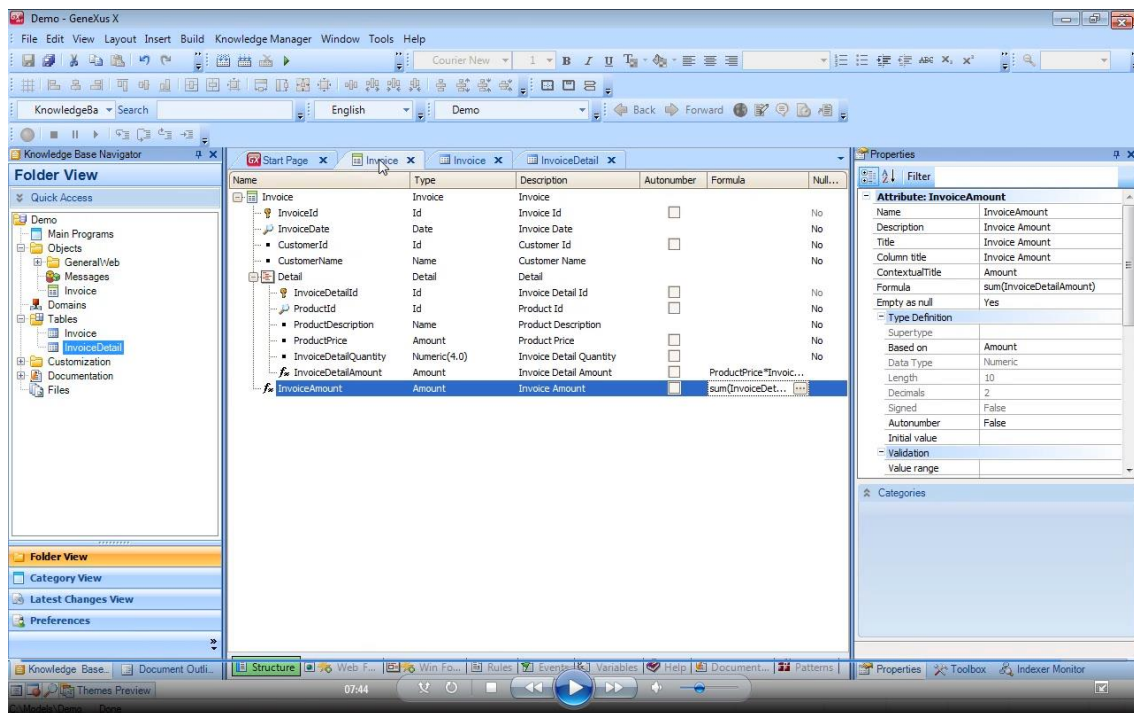
Salvamos. A partir de la **estructura** de este objeto **transacción**, Genexus obtendrá la información para crear automáticamente en la base de datos las **tablas** necesarias para almacenar los datos de las facturas. Por ejemplo vemos con esta transacción se van a crear 2 tablas.



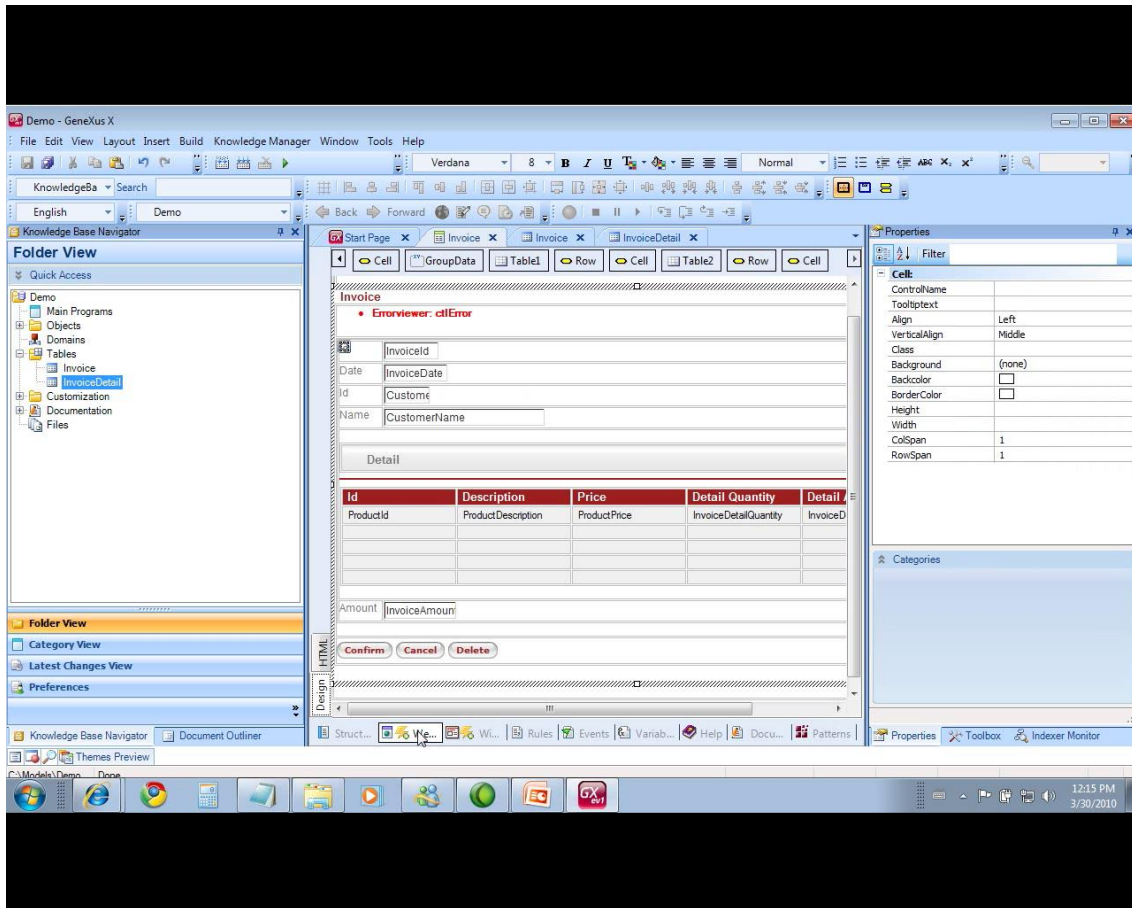
Una tabla **Invoice** correspondiente al cabezal de la factura y



una tabla **InvoiceDetail** correspondiente a las líneas de detalle de la factura. Otra cosa interesante a tomar en cuenta



es que a partir de la información que ingresamos en la transacción y dado que nuestra aplicación es una aplicación web, Genexus nos va a generar automáticamente



el formulario **web** con todos los controles necesarios para insertar, editar y eliminar facturas. A continuación GeneXus construirá la aplicación web para facturación sin que tengamos que programar ni una sola línea.