Artículo
Estevan Martinez · 5 jun, 2019 Lectura de 7 min

# El arte de mapear Globals para Clases (1 de 3)

¿Alguno de ustedes desea traer de vuelta una antigua aplicación MUMPS? Siga estos pasos para mapear sus globales ya establecidos para clases y mostrar todos esos hermosos datos en Objects y SQL

Al seguir los sencillos pasos en este artículo y los dos siguientes, podrá mapear todas las variables globales, excepto las más difíciles, para clases de Caché. Para los que estén locos pondré un archivo zip con diferentes mapeos que he coleccionado a lo largo de los años. Esto NO es para datos nuevos; si no cuenta con datos globales ya establecidos, por favor, utilice el almacenamiento predeterminado.

Si sus variables globales todavía no tienen ni pies ni cabeza, envíe algunos ejemplos de sus datos a Support@InterSystems.com y estaremos encantados de ayudarle a resolver su problema.

Pasos para mapear una variable global para una clase:

1. Identifique algún patrón que se repita en los datos globales
2. Identifique qué la hace una clave única
3. Identifique las propiedades y sus categorías
4. Defina las propiedades en la clase (no olvide las propiedades de los subíndices de las variables)
5. Defina el índice IdKey
6. Establezca la definición de almacenamiento:
    a. Establezca los subíndices, e incluya la IdKey
    b. Establezca una sección para los datos
    c. Ignore la sección "Row ID (Identificación de la fila)". El 99% de las veces el valor que desea es el predeterminado, así que deje que el sistema lo llene por usted
7. Compile y pruebe su clase / tabla

**¡Veamos un ejemplo!**

Digamos que tiene 2 globales que se parecen a esto:

```
^mapping("Simple",1)="Brendan Bannon^55192^SQL Manager"
^mapping("Simple",2)="Nicole Aaron^63375^Support Specialist"
^mapping("Simple",3)="Kyle Baxter^61155^Senior Support Specialist"
^mapping("Simple",4)="Prasad Kari^58471^Support Specialist"
^mapping("Simple",5)="Clive Morgan^57982^Senior Support Specialist"

^index("Simple","HireDate",55192,1)=""
^index("Simple","HireDate",57982,5)=""
^index("Simple","HireDate",58471,4)=""
^index("Simple","HireDate",61155,3)=""
^index("Simple","HireDate",63375,2)=""
```

Repasemos los 7 pasos que se deben realizar para crear una clase que nos permita ver estos datos mediante SQL y Objects:

**Paso 1**

El ^mapping global es tan sencillo como parece. Cada nodo se ve como si tuviera el mismo tipo de datos. Nos

ocuparemos del ^index global cuando terminemos con el ^mapping global.

**Paso 2**

El primer subíndice solo es una constante. El segundo subíndice es un contador aumentado, que parece ser único para cada fila de datos.

**Paso 3**

Parece que las propiedades podrían ser: Name, HireDate, Title. Sin mirar el ^index global o sin conocer a alguien que sepa cómo se utilizan los datos, sería difícil saber que 55192 es una fecha, y mucho menos que es una fecha de contratación. No olvide definir una propiedad para el valor en el segundo subíndice.

**Paso 4**

```
Property Name As %String;
Property HireDate As %Date;
Property Title As %String;
Property Sub2 As %Integer;
```

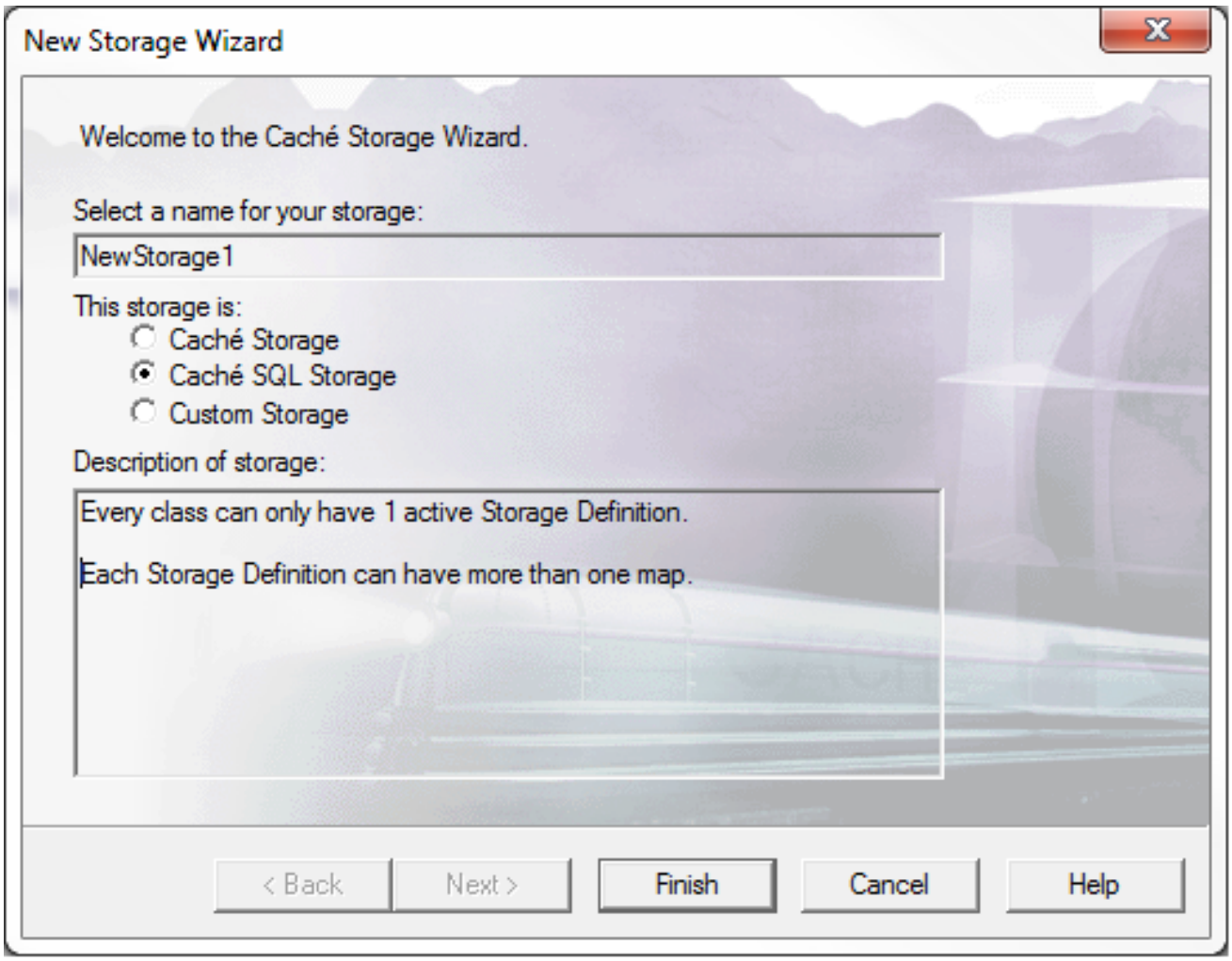
**Paso 5**

```
Debe definir un índice IdKey para cada clase que utilice el Caché SQL Storage
        Index Master On Sub2 [ IdKey ];
No es necesario definir un índice para los mapas, pero es una buena idea hacerlo.
        Index hireDateindex On HireDate;
```

**Paso 6**

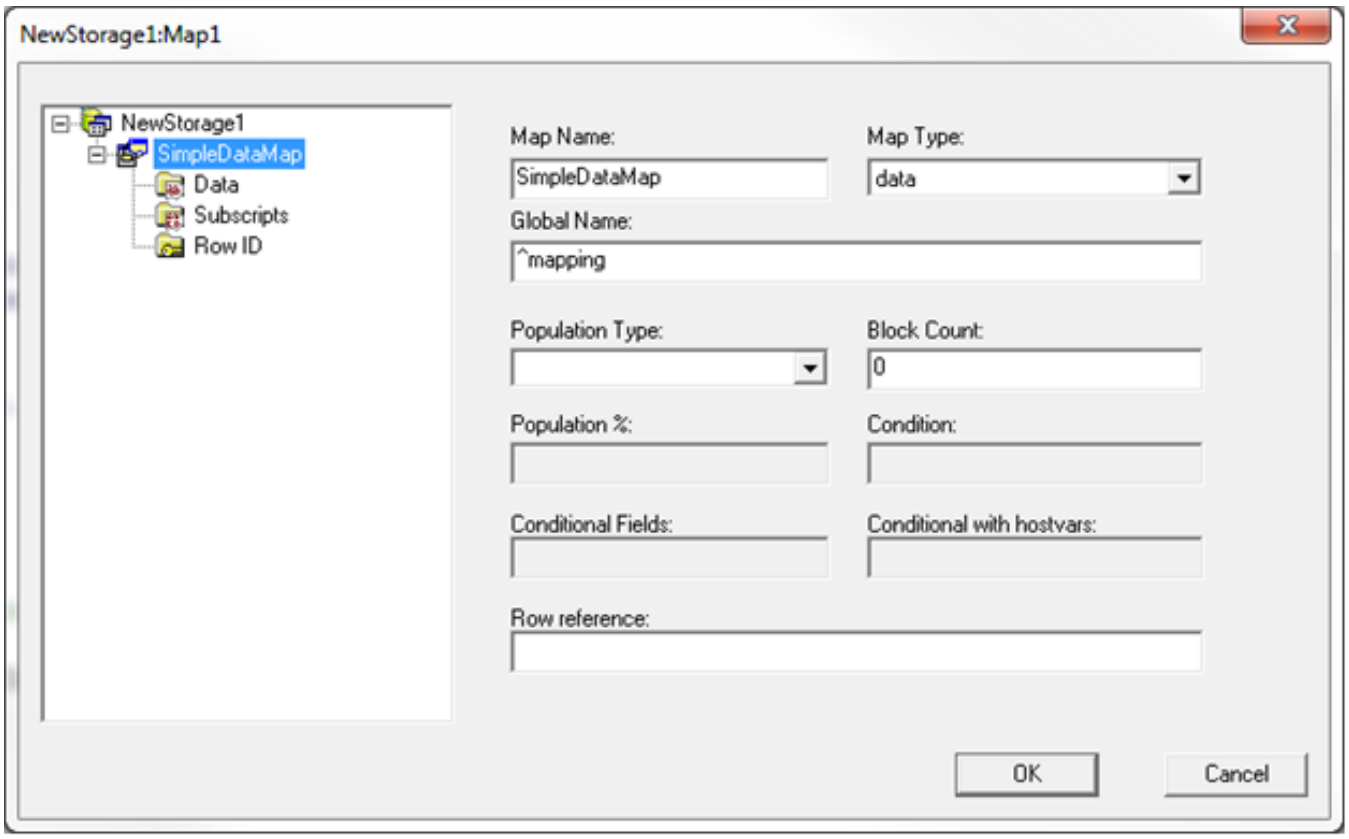
Ahora necesitamos crear el mapeo entre la clase y la variable global. Me gusta utilizar el asistente para hacerlo, así que solamente mostraré las capturas de pantalla, pero si usted es bueno escribiendo en xml puede establecer manualmente la definición del Caché SQL Storage.

Haga clic sobre el icono "Storage". Puede llamar a la definición de almacenamiento las veces que quiera, yo utilicé la definición predeterminada. Haga clic en "Caché SQL Storage" y posteriormente en "Finish".



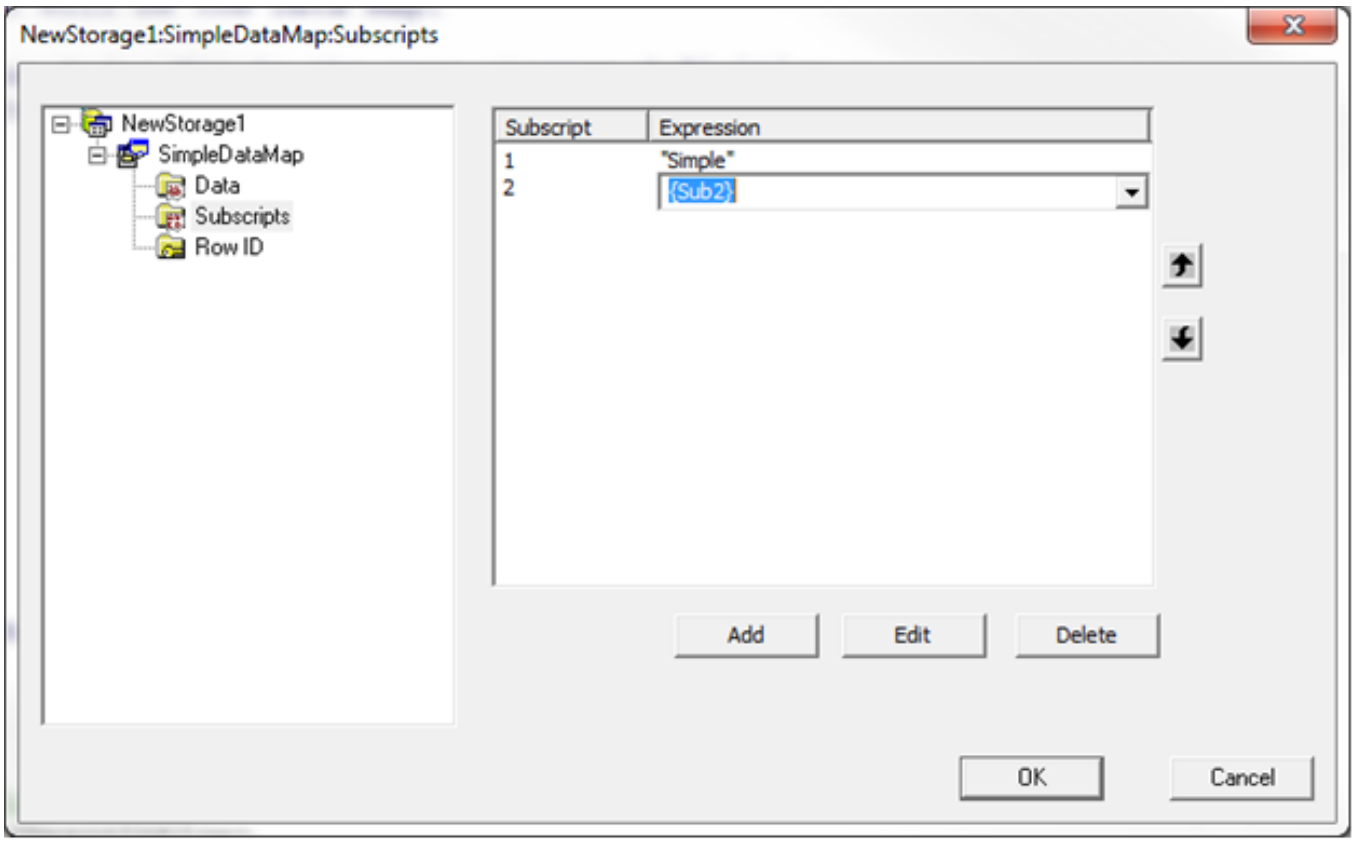

En la ventana que se muestra a continuación, cambié el campo "Map Name", pero Map1 también funciona bien. Lo único que debe completar es la casilla de "Global Name". No olvide colocar el"^". En cada clase permanente necesita tener un mapa de datos. Puede tener varios índices para los mapas.

Al hacer clic en OK, saldrá del asistente, así que aléjese de esta opción hasta que termine de utilizarlo.



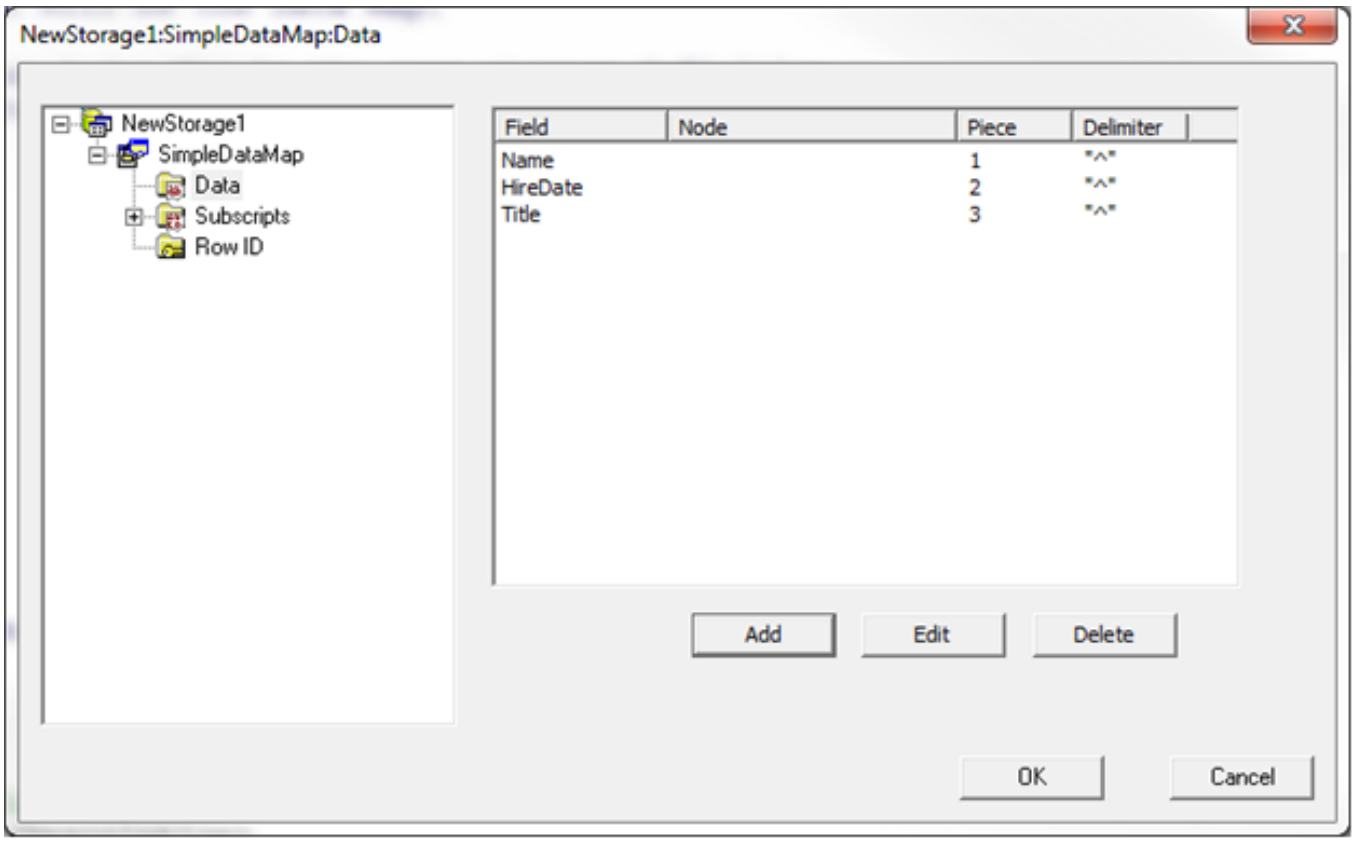

**Paso 6a**

Lo siguiente que quiero hacer son subíndices. Aquí definiremos todos los subíndices de las variables globales, incluso a IdKey (IdKey podría ser más que un subíndice). En la casilla de la expresión, se hace referencia a un campo al ponerlo entre estos símbolos {}. Puede colocar cualquier expresión COS que sea válida aquí. Mantendremos las cosas sencillas, de modo que únicamente tenemos una constante, Subscript 1 y un campo en Subscript 2.



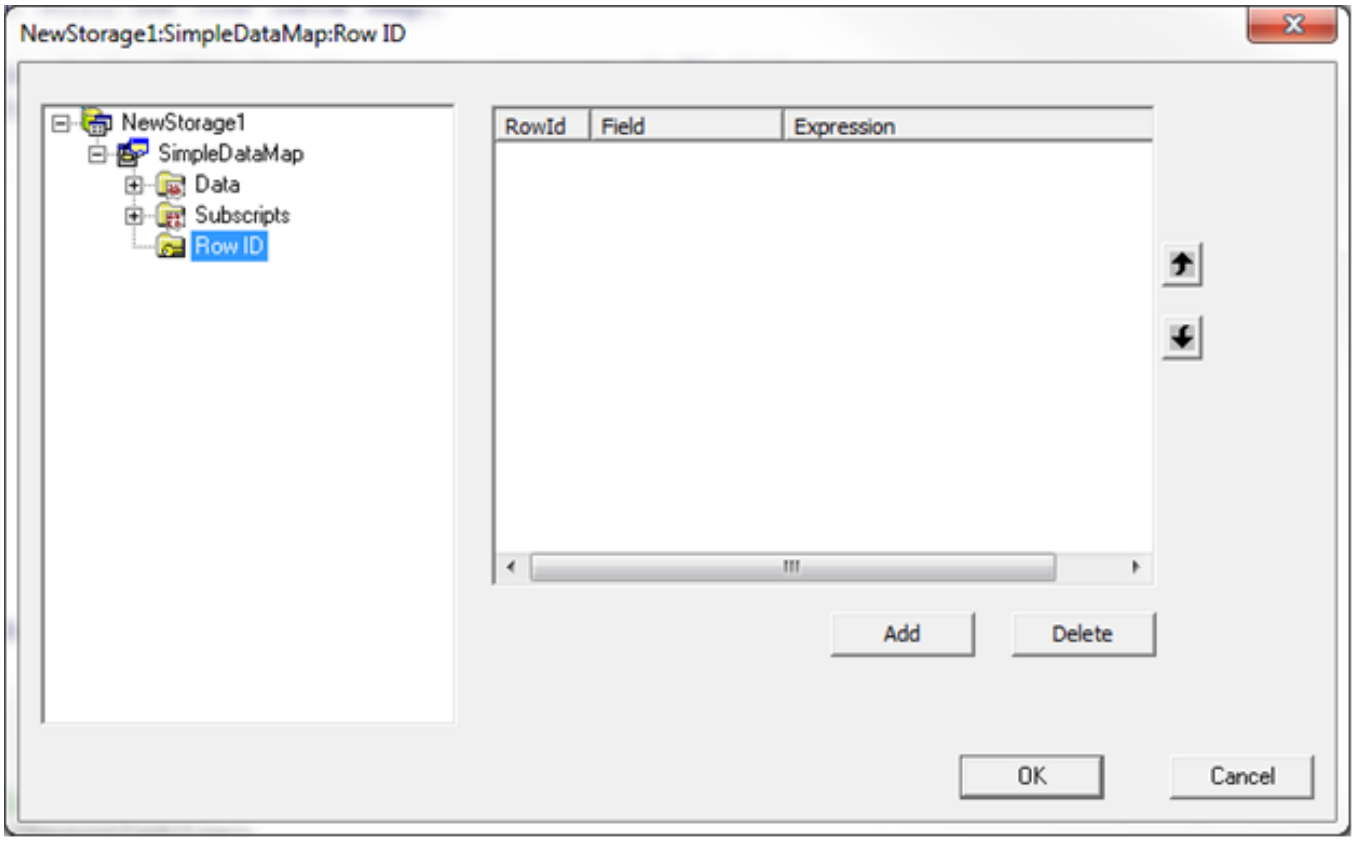

**Paso 6b**

Para los datos, describimos cualquier subíndice que se coloque después de la IdKey, más todos los datos que se encuentren a la derecha del signo igual. La columna "Node" se utilizará para colocar cualquier subíndice adicional (en mi siguiente post habrá un ejemplo de esto). "Piece" y "Delimiter" describen la ubicación de las propiedades en la variable global. Para el valor predeterminado se utiliza el comando $PIECE, cuya finalidad es analizar los datos de la variable global.

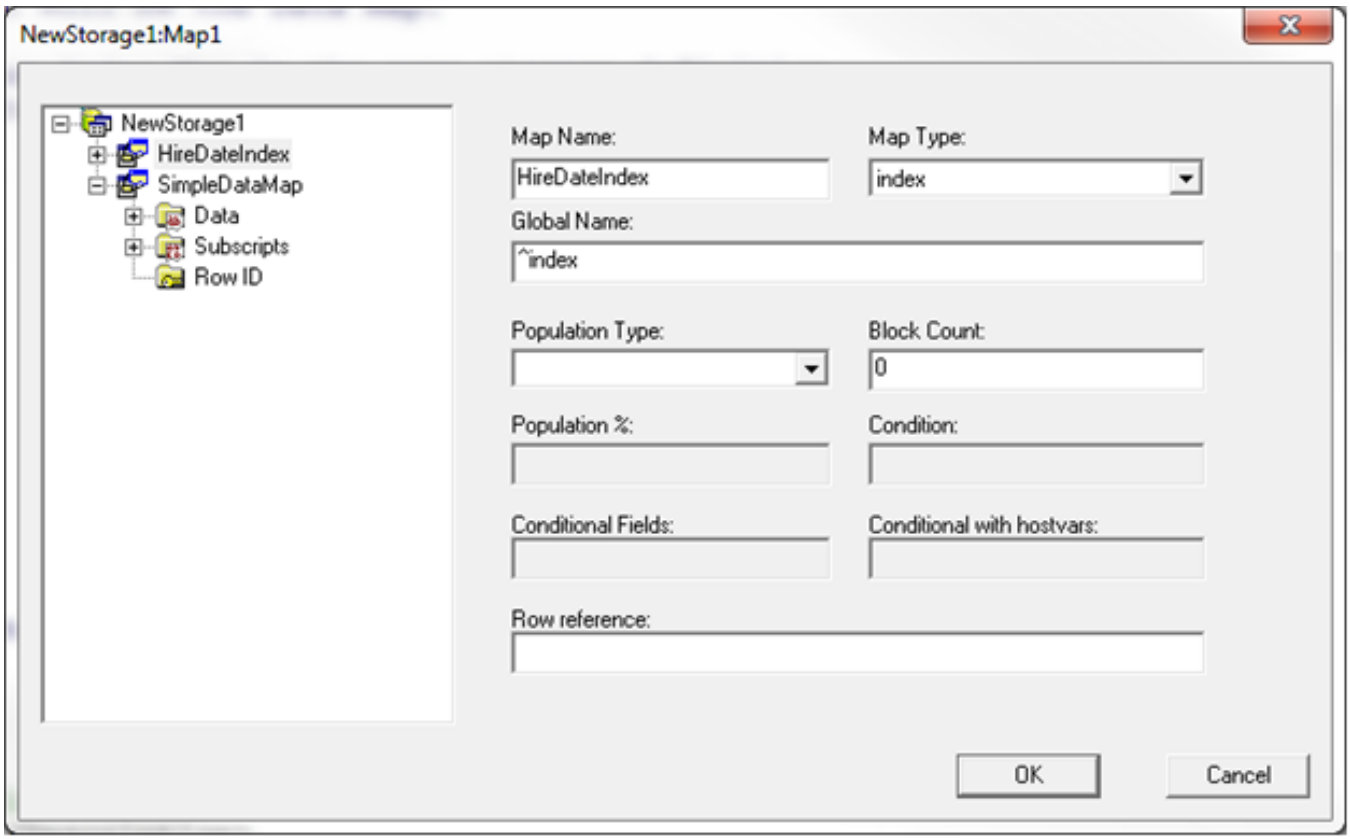



**Paso 6c**

Aquí no verá nada, le pedí que dejara el espacio en blanco.

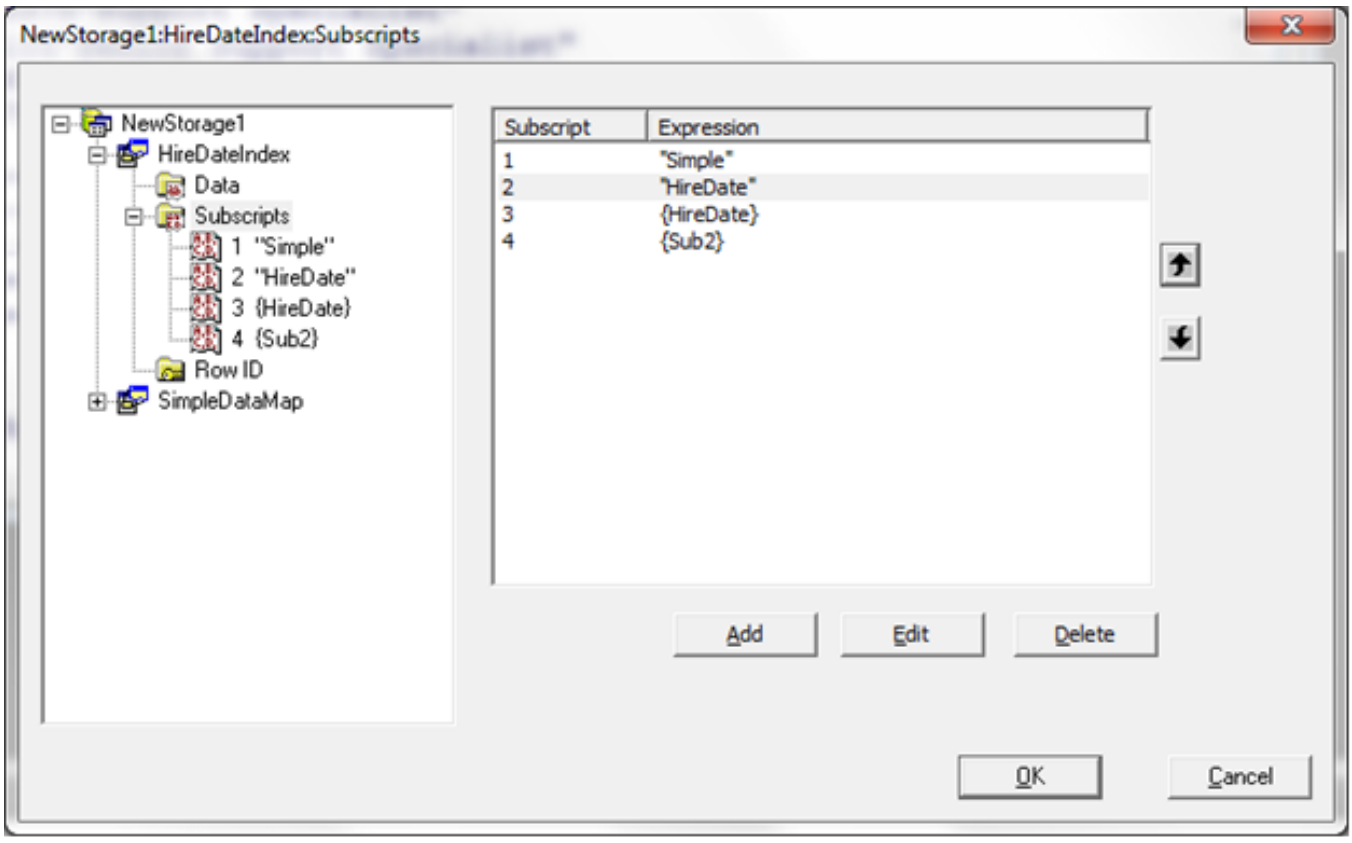



Ahora repita el paso 6 para el ^index global.



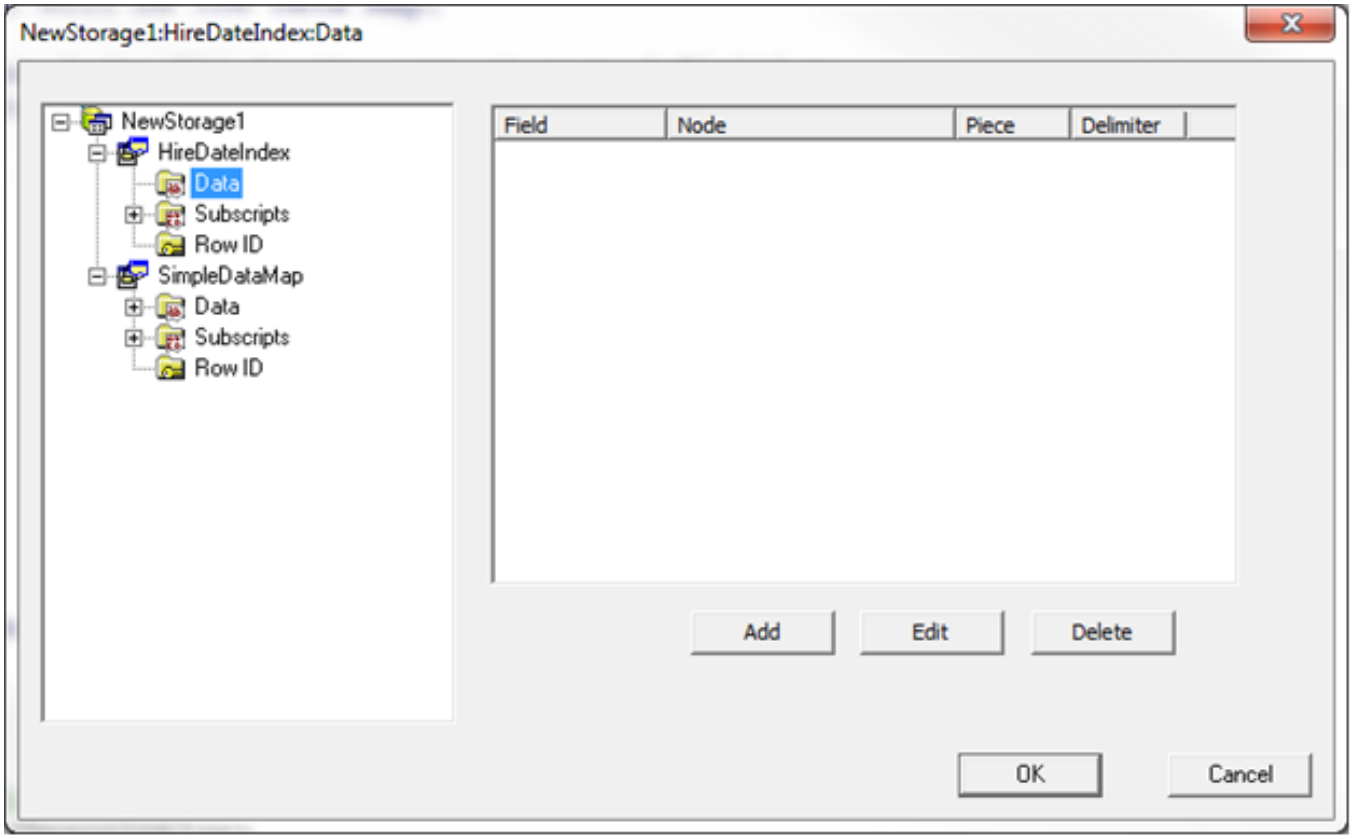

**Paso 6a**

Esta vez se muestran cuatro subíndices: dos de ellos son constantes, HireDate e IdKey (Subíndice 2). Cada índice del mapa debe tener la posibilidad de construir la IdKey. La mayoría de las veces estará en los subíndices, pero también podría ser parte de los datos.



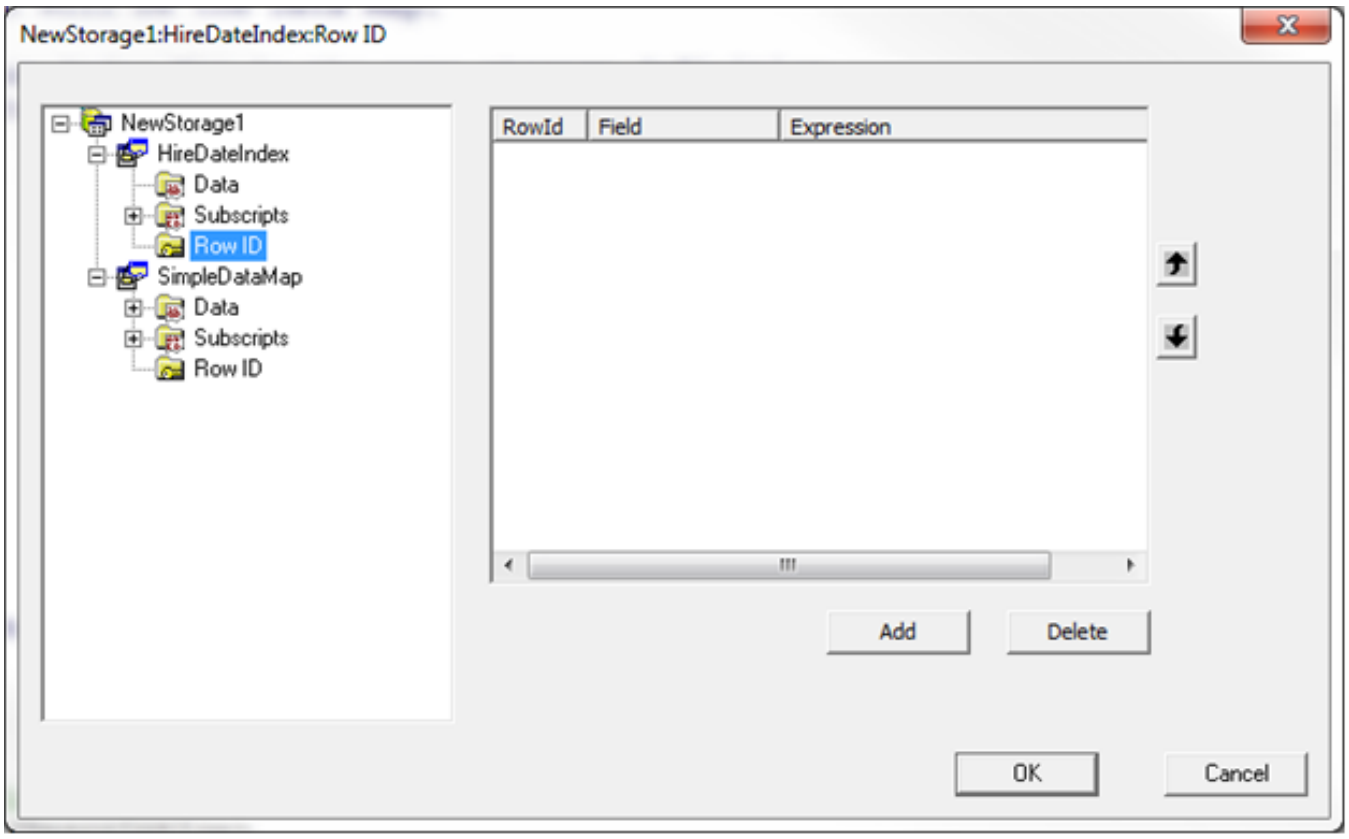
**Paso 6b**

No existen subíndices adicionales ni datos, por lo que el espacio se encuentra en blanco




**Paso 6c**

Todavía no hay nada que hacer aquí.




**Paso 7**

Lo único que falta es compilar la clase y después trataremos de consultar la tabla para asegurarnos de que los datos se muestran correctamente.

```
Compilation started on 08/15/2016 15:20:58 with qualifiers 'fck /checkuptodate=expandedonly'
Compiling class Mapping.Example1
Compiling table Mapping.Example1
Compiling routine Mapping.Example1.1
Compilation finished successfully in 0.270s.
```

```
SELECT Sub2, Name, HireDate, Title FROM Mapping.Example1
```

| Sub2 | Name | HireDate | Title |
|---|---|---|---|
| 1 | Brendan Bannon | 1992-02-10 | SQL Manager |
| 2 | Nicole Aaron | 2014-07-07 | Support Specialist |
| 3 | Kyle Baxter | 2008-06-08 | Senior Support Specialist |
| 4 | Prasad Kari | 2001-02-01 | Support Specialist |
| 5 | Clive Morgan | 1999-10-01 | Senior Support Specialist |

Y para aquellos a los que les gusta escribir, aquí se puede ver la definición de almacenamiento en xml

```
/// Cada clase tiene solo una definición de almacenamiento activa.
/// Cada definición de almacenamiento puede tener más de un mapa.
Storage NewStorage1
```

{

<SQLMap name="HireDateIndex">
<ConditionalWithHostVars></ConditionalWithHostVars>
<Global>^index</Global>
<Subscript name="1">
<Expression>"Simple"</Expression>
</Subscript>
<Subscript name="2">
<Expression>"HireDate"</Expression>
</Subscript>
<Subscript name="3">
<Expression>{HireDate}</Expression>
</Subscript>
<Subscript name="4">
<Expression>{Sub2}</Expression>
</Subscript>
<Type>index</Type>
</SQLMap>
<SQLMap name="SimpleDataMap">
<Data name="HireDate">
<Delimiter>"^"</Delimiter>
<Piece>2</Piece>
</Data>
<Data name="Name">
<Delimiter>"^"</Delimiter>
<Piece>1</Piece>
</Data>
<Data name="Title">
<Delimiter>"^"</Delimiter>
<Piece>3</Piece>
</Data>
<Global>^mapping</Global>
<Subscript name="1">
<Expression>"Simple"</Expression>
</Subscript>
<Subscript name="2">
<Expression>{Sub2}</Expression>
</Subscript>
<Type>data</Type>
</SQLMap>
<StreamLocation>^Mapping.Example1S</StreamLocation>
<Type>%CacheSQLStorage</Type>
}

Para aquellos que no quieren escribir, aquí hay un archivo con las variables globales y la clase:
mappingexample1.zip

Si desea obtener más información, consulte la Parte 2

#Globals #Mapeo #SQL #Caché