

Fwk.Configuration

BLOQUE DE CONFIGURACIÓN

El bloque de configuración que provee el Framework de Fwk permite a las aplicaciones (Back-End o Front-End) disponer un conjunto de mecanismos que facilitan el acceso a archivos o bases de datos de configuración.

Existen casos en que las distintas aplicaciones necesitaran persistir cierta información en algún lugar de acceso rápido que no sea la base de datos del sistema ni tampoco el archivo .config.

Básicamente el sistema de configuración consiste en una serie de componentes del framework fwk que proveen de los archivos o base de datos de configuración.

La estructura de una configuración vista como xml podría ser:

Estructura:

```
<ConfigurationFile>
  <Groups>
    <Group name="NombreGrupo_Uno">
      <Keys>
        <Key name="NombrePropiedad_Uno" >
          <Value><![CDATA[ValorPropiedad_Uno]]></Value>
        </Key>
        <Key name="NombrePropiedad_Dos" >
          <Value><![CDATA[ValorPropiedad_Dos]]></Value>
        </Key>
      </Keys>
    </Group>
    <Group name="NombreGrupo_Dos">
      .
      .
      .
    </Group>
  </Groups>
</ConfigurationFile>
```

Código 1.0

Grupos

Los archivos de configuración como podemos apreciar están divididos en “**Grupos**” donde cada uno de ellos es un repositorio de ciertas configuraciones que pertenecen a un mismo nivel de abstracción. Por ejemplo podemos tener un grupo llamado “*FuntionalExceptions*” donde los elementos que la compongan sean todos mensajes de errores de tipo funcional o de negocio que mostrara una aplicaron en las pantallas de los clientes. Por otro lado podemos tener otro grupo que contenga información de cadenas de conexión a distintas bases de datos o rutas de acceso a archivos de recursos.

Keys

Los distintos grupos que conforman una archivo de configuración como vimos están formados por uno o mas elementos estos elementos son “**Keys**” y están identificados por un atributo único dentro del grupo llamado “**name**”.

Valor

El valor de estas Keys o claves se encuentra el nodo “**Value**” dentro de un **CDATA[]**. Este valor es el que realmente le interesa a aquella aplicaron que la solicite.

Aplicaciones Clientes o consumidoras de configuracion

Estos son los consumidores del servicio de configuración. Los clientes se utilizan *ConfigurationMannager* para tal propósito este servicio de Windows puede ser Local o Remoto depende del diseño de arquitectura de la solución que se establezca.

Las distintas aplicaciones pueden ser alguno de los siguientes:

- Servicios de Windows
- Aplicaciones cliente
- Despachadores de servicios
- Etc.

La ubicación del bloque de configuración es en Fwk.configuration y su componente más importante es el ConfigurationManager. Este es el único que utilizara el desarrollador para interactuar con el bloque de configuración.

- Interfaz publica de ConfigurationMannager:

Método	Descripción
GetConfigurationFile	Obtiene un ConfigurationFile con la estructura vista en Codigo 10 Parámetros: <ul style="list-style-type: none">▪ providerName: Nombre del proveedor de configuración utilizado
GetFileVersionStatus	
GetProperty	Devuelve un grupo de un BaseConfigFile o archivo de configuración del catalogo. Parámetros: <ul style="list-style-type: none">▪ providerName: Nombre del proveedor de configuración utilizado▪ groupName: Nombre del grupo donde se encuentra la propiedad a obtener

	<ul style="list-style-type: none">▪ propertyName: Nombre de la propiedad que se desea obtener
GetGroup	Devuelve un grupo de un BaseConfigFile Parámetros: <ul style="list-style-type: none">▪ providerName: Nombre del proveedor de configuración utilizado▪ groupName: Nombre del grupo donde se encuentra la propiedad a obtener.

Tabla 1.0

Proveedores de configuración

Los proveedores de configuración son los que dan origen a la configuración propiamente dicha. Estos tienen por objetivo principal Identificar el origen de dato de las configuraciones.

Propiedades de un proveedor

Propiedad	Descripción y valores
name	Nombre que identifica el proveedor. Este valor es que utiliza el ConfigurationMannaget para trabajar con una configuración determinada
type	Indica el tipo de origen de almacén de la información. Hasta el momento existen : <ul style="list-style-type: none">▪ xml La información está contenida en un archivo xml▪ sqldatabase: La información está contenida en una tabla de sql cuya estructura esta dado por los script de instalación de bases de datos del framework
sourceinfo	Identifica diferentes orígenes de información según el tipo de proveedor <ul style="list-style-type: none">• Si type = "xml" → "ignore sourceinfo"• Si type = "sqldatabase" → nombre de cadena de conexión
baseConfigFile	Si type = "xml" Nombre de archivo físico de configuracion XML si Si type = "sqldatabase" identificador de configuracion

Tabla 2.0

Almacén según el tipo de de proveedor

El almacenamiento de la información de configuración depende del tipo de de proveedor. Según el atributo **type** en la tabla 2.0. puede ser en un archivo o una base de datos.

A continuación se detallan ambos casos.

- **Archivos de Configuración provider type xml**

Por ejemplo ConfigurationSample.xml

Estos archivos contienen valores de configuración de cada aplicación, organizado en grupos con sus nombres, propiedades para cada uno y sus respectivos valores.

El Código 2.0 nos muestra un ejemplo práctico de cómo se podría ver un archivo xml de estas características:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<ConfigurationFile>
  <Groups>
    <Group name="FuntionalsExceptions">
      <Keys>
        <Key name="Excepcion1" encrypted="false">
          <Value><![CDATA[Error 2]]></Value>
        </Key>
        <Key name="Excepcion2" encrypted="false">
          <Value><![CDATA[Error 2]]></Value>
        </Key>
      </Keys>
    </Group>
    <Group name="ClientMessages">
      <Keys>
        <Key name="Message1" encrypted="false">
          <Value><![CDATA[Hola mundo.]]></Value>
        </Key>
        <Key name=" Message2" encrypted="false">
          <Value><![CDATA[Ejemplo de fwk de configuracion.]]></Value>
        </Key>
      </Keys>
    </Group>
  </Groups>
</ConfigurationFile>
```

Código 2.0

- **Tablas sql de Configuración provider type sqldatabase**

Esta es otra opción válida para almacenar todas las configuraciones que deseemos. Si nuestro proveedor es sqldatabase entonces podremos cargar y recuperar nuestras configuraciones en una tabla de SQL Server

La estructura de la tabla es la siguiente.

Nombre: fwk_ConfigMannager

ConfigurationFileName	nvarchar(50)
[group]	nvarchar(50)
[key]	nvarchar(50)
encrypted	bit
value	nvarchar(1000)

Para este tipo de proveedores es necesario que exista una base de datos con una tabla cuya estructura este representada por el siguiente script

Fwk_ConfigMannager.sql

Procedimiento de configuración en la aplicación

A continuación se detalla el procedimiento para configurar una la aplicación que utilice el sistema de configuración o bloque fwk.configuration

1. Abra el archivo App.config o NombreEjecutable.exe.Config con cualquier editor de texto.
2. Agregue al archivo la sección **FwkConfigProvider** en **configSections**

```
<configSections>
  <section name="FwkConfigProvider" type="Fwk.ConfigSection.ConfigProviderSection, Fwk.Bases" />
</configSections>
```

3. Agregue la sección con el/los providers de configuración y establezca un proveedor por defecto

```
<!--Configuracion del los proveedores de configuracion del framework -->
<FwkConfigProvider defaultProviderName="p1">
  <Providers>
    <add name="p1" type="local" baseConfigFile="ConfigurationManager.xml" />
    <add name="p2" type="sqldatabase " sourceinfo ="test" baseConfigFile
      ="bgbang" />
  </Providers>
</FwkConfigProvider>
```

Código 2.0

El siguiente código muestra un ejemplo práctico de cómo debería verse el archivo de configuración:

```
<configuration>
  <configSections>
    <section name="FwkConfigProvider" type="Fwk.ConfigSection.ConfigProviderSection,
Fwk.Bases" />
  </configSections>

  <connectionStrings>

    <add name="test"
      connectionString="Database=fwktest;Server=(local)\SQLEXPRESS;Integrated
      Security=SSPI" providerName="System.Data.SqlClient" />
  </connectionStrings>

  <!--Configuracion del los proveedores de configuracion del framework -->
  <FwkConfigProvider defaultProviderName="p1">
    <Providers>
      <add name="p1" type="local" baseConfigFile="ConfigurationManager.xml"
appid="" />
      <add name="p2" type="sqldatabase " sourceInfo ="test" baseConfigFile
      ="bgbang" />
    </Providers>
  </FwkConfigProvider>
</configuration>
```

Código 3.0