

Re: Typed DataSets in VS2005

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2007-03/msg04631.html>

- *From:* "Peter Bradley" <pbradley@xxxxxxxxxxx>
 - *Date:* Thu, 29 Mar 2007 14:13:00 +0100
-

If it was just for its internal use, I wouldn't be bothered. I've tried it out, and there is no way of changing that setting via a configuration file once the dll has been deployed (and put in the GAC in our case) that has worked for me. If you can suggest a way, I'd be very grateful.

Peter

"Champika Nirosh" <test@xxxxxx> wrote in message
news:%23SfzONgcHHA.2268@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

So now let's come back again so the designer keep it in the setting.Designer.cs for its internal use.. so what bother you there?

Nirosh.

"Peter Bradley" <pbradley@xxxxxxxxxxx> wrote in message
news:%23BTOVHgcHHA.4260@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

Just in case anyone else has any doubts, here's the generated xsd file (slightly anonymised):

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema id="ItregMailDetailsTypedDataSet"
targetNamespace="http://tempuri.org/ItregMailDetailsTypedDataSet.xsd
xmlns:mstns="http://tempuri.org/ItregMailDetailsTypedDataSet.xsd
xmlns="http://tempuri.org/ItregMailDetailsTypedDataSet.xsd
xmlns:xs="http://www.w3.org/2001/XMLSchema
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata"
xmlns:msprop="urn:schemas-microsoft-com:xml-msprop"
attributeFormDefault="qualified" elementFormDefault="qualified">
<xs:annotation>
<xs:appinfo source="urn:schemas-microsoft-com:xml-msdatasource">
<DataSource DefaultConnectionIndex="0"
FunctionsComponentName="QueriesTableAdapter" Modifier="AutoLayout,
AnsiClass, Class, Public" SchemaSerializationMode="IncludeSchema"
xmlns="urn:schemas-microsoft-com:xml-msdatasource">
<Connections>
```

Re: Typed DataSets in VS2005

```
<Connection AppSettingsObjectName="Settings"
AppSettingsPropertyName="aservernamedevConnectionString"
ConnectionStringObject="" IsAppSettingsProperty="True"
Modifier="Assembly" Name="aservernamedevConnectionString (Settings)"
ParameterPrefix="@ "
PropertyReference="ApplicationSettings.ItregMailDetailsTypedDataSet.Properties.Settings.GlobalRe
Provider="System.Data.SqlClient">
</Connection>
</Connections>
<Tables>
<TableAdapter BaseClass="System.ComponentModel.Component"
DataAccessorModifier="AutoLayout, AnsiClass, Class, Public"
DataAccessorName="atablenameTableAdapter"
GeneratorDataComponentClassName="atablenameTableAdapter"
Name="atablename"
UserDataComponentName="atablenameTableAdapter">
<MainSource>
<DbSource ConnectionRef="aservernamedevConnectionString
(Settings)" DbObjectName="aservernamedev.dbo.atablename"
DbObjectType="Table" FillMethodModifier="Public"
FillMethodName="Fill"
GenerateMethods="Both" GenerateShortCommands="True"
GeneratorGetMethodName="GetData" GeneratorSourceName="Fill"
GetMethodModifier="Public" GetMethodName="GetData"
QueryType="Rowset"
ScalarCallRetval="System.Object, mscorlib, Version=2.0.0.0,
Culture=neutral, PublicKeyToken=b77a5c561934e089"
UseOptimisticConcurrency="True" UserGetMethodName="GetData"
UserSourceName="Fill">
<DeleteCommand>
<DbCommand CommandType="Text" ModifiedByUser="False">
<CommandText>DELETE FROM [dbo].[atablename] WHERE
([mailbox] = @Original mailbox) AND ([storage_group] =
@Original storage_group) AND ([server] =
@Original server)</CommandText>
<Parameters>
<Parameter AllowDBNull="False" AutogeneratedName=""
DataSourceName="" DbType="AnsiString" Direction="Input"
ParameterName="@Original mailbox" Precision="0"
ProviderType="VarChar"
Scale="0" Size="0" SourceColumn="mailbox"
SourceColumnNullMapping="False"
SourceVersion="Original">
</Parameter>
<Parameter AllowDBNull="False" AutogeneratedName=""
DataSourceName="" DbType="AnsiString" Direction="Input"
ParameterName="@Original storage_group" Precision="0"
ProviderType="VarChar" Scale="0" Size="0"
SourceColumn="storage_group"
SourceColumnNullMapping="False" SourceVersion="Original">
</Parameter>
```

Re: Typed DataSets in VS2005

```
<Parameter AllowDBNull="False" AutogeneratedName=""  
DataSourceName="" DbType="AnsiString" Direction="Input"  
ParameterName="@Original_server" Precision="0"  
ProviderType="VarChar"  
Scale="0" Size="0" SourceColumn="server"  
SourceColumnNullMapping="False"  
SourceVersion="Original">  
</Parameter>  
</Parameters>  
</DbCommand>  
</DeleteCommand>  
<InsertCommand>  
<DbCommand CommandType="Text" ModifiedByUser="False">  
<CommandText>INSERT INTO [dbo].[atablename]  
([mailbox], [storage_group], [server]) VALUES (@mailbox,  
@storage_group,  
@server);  
SELECT mailbox, storage_group, server FROM atablename WHERE  
(mailbox =  
@mailbox)</CommandText>  
</Parameters>  
<Parameter AllowDBNull="False" AutogeneratedName=""  
DataSourceName="" DbType="AnsiString" Direction="Input"  
ParameterName="@mailbox" Precision="0" ProviderType="VarChar"  
Scale="0"  
Size="0" SourceColumn="mailbox" SourceColumnNullMapping="False"  
SourceVersion="Current">  
</Parameter>  
<Parameter AllowDBNull="False" AutogeneratedName=""  
DataSourceName="" DbType="AnsiString" Direction="Input"  
ParameterName="@storage_group" Precision="0" ProviderType="VarChar"  
Scale="0" Size="0" SourceColumn="storage_group"  
SourceColumnNullMapping="False" SourceVersion="Current">  
</Parameter>  
<Parameter AllowDBNull="False" AutogeneratedName=""  
DataSourceName="" DbType="AnsiString" Direction="Input"  
ParameterName="@server" Precision="0" ProviderType="VarChar"  
Scale="0"  
Size="0" SourceColumn="server" SourceColumnNullMapping="False"  
SourceVersion="Current">  
</Parameter>  
</Parameters>  
</DbCommand>  
</InsertCommand>  
<SelectCommand>  
<DbCommand CommandType="Text" ModifiedByUser="False">  
<CommandText>SELECT mailbox, storage_group, server  
FROM dbo.atablename</CommandText>  
</Parameters>  
</Parameters>  
</DbCommand>
```

Re: Typed DataSets in VS2005

```
</SelectCommand>
<UpdateCommand>
<DbCommand CommandType="Text" ModifiedByUser="False">
<CommandText>UPDATE [dbo].[atablename] SET [mailbox] =
@mailbox, [storage_group] = @storage_group, [server] = @server WHERE
(([mailbox] = @Original_mailbox) AND ([storage_group] =
@Original_storage_group) AND ([server] = @Original_server));
SELECT mailbox, storage_group, server FROM atablename WHERE
(mailbox =
@mailbox)</CommandText>
<Parameters>
<Parameter AllowDBNull="False" AutogeneratedName=""
DataSourceName="" DbType="AnsiString" Direction="Input"
ParameterName="@mailbox" Precision="0" ProviderType="VarChar"
Scale="0"
Size="0" SourceColumn="mailbox" SourceColumnNullMapping="False"
SourceVersion="Current">
</Parameter>
<Parameter AllowDBNull="False" AutogeneratedName=""
DataSourceName="" DbType="AnsiString" Direction="Input"
ParameterName="@storage_group" Precision="0" ProviderType="VarChar"
Scale="0" Size="0" SourceColumn="storage_group"
SourceColumnNullMapping="False" SourceVersion="Current">
</Parameter>
<Parameter AllowDBNull="False" AutogeneratedName=""
DataSourceName="" DbType="AnsiString" Direction="Input"
ParameterName="@server" Precision="0" ProviderType="VarChar"
Scale="0"
Size="0" SourceColumn="server" SourceColumnNullMapping="False"
SourceVersion="Current">
</Parameter>
<Parameter AllowDBNull="False" AutogeneratedName=""
DataSourceName="" DbType="AnsiString" Direction="Input"
ParameterName="@Original_mailbox" Precision="0"
ProviderType="VarChar"
Scale="0" Size="0" SourceColumn="mailbox"
SourceColumnNullMapping="False"
SourceVersion="Original">
</Parameter>
<Parameter AllowDBNull="False" AutogeneratedName=""
DataSourceName="" DbType="AnsiString" Direction="Input"
ParameterName="@Original_storage_group" Precision="0"
ProviderType="VarChar" Scale="0" Size="0"
SourceColumn="storage_group"
SourceColumnNullMapping="False" SourceVersion="Original">
</Parameter>
<Parameter AllowDBNull="False" AutogeneratedName=""
DataSourceName="" DbType="AnsiString" Direction="Input"
ParameterName="@Original_server" Precision="0"
ProviderType="VarChar"
Scale="0" Size="0" SourceColumn="server"
```

Re: Typed DataSets in VS2005

```
SourceColumnNullMapping="False"  
SourceVersion="Original">  
</Parameter>  
</Parameters>  
</DbCommand>  
</UpdateCommand>  
</DbSource>  
</MainSource>  
<Mappings>  
<Mapping SourceColumn="mailbox" DataSetColumn="mailbox" />  
<Mapping SourceColumn="storage_group"  
DataSetColumn="storage_group" />  
<Mapping SourceColumn="server" DataSetColumn="server" />  
</Mappings>  
<Sources>  
<DbSource ConnectionRef="aservernamedevConnectionString  
(Settings)" DbObjectName="aservernamedev.dbo.usp_itreg_roundrobin"  
DbType="StoredProcedure" FillMethodModifier="Public"  
FillMethodName="FillBy" GenerateMethods="Both"  
GenerateShortCommands="True" GeneratorGetMethodName="GetDataBy"  
GeneratorSourceName="FillBy" GetMethodModifier="Public"  
GetMethodName="GetDataBy" QueryType="Rowset"  
ScalarCallRetval="System.Object, mscorlib, Version=2.0.0.0,  
Culture=neutral, PublicKeyToken=b77a5c561934e089"  
UseOptimisticConcurrency="True" UserGetMethodName="GetDataBy"  
UserSourceName="FillBy">  
<SelectCommand>  
<DbCommand CommandType="StoredProcedure"  
ModifiedByUser="False">  
<CommandText>dbo.usp_itreg_roundrobin</CommandText>  
<Parameters>  
<Parameter AllowDBNull="True" AutogeneratedName=""  
DataSourceName="" DbType="Int32" Direction="ReturnValue"  
ParameterName="@RETURN_VALUE" Precision="10"  
ProviderType="Int" Scale="0"  
Size="4" SourceColumnNullMapping="False" SourceVersion="Current">  
</Parameter>  
</Parameters>  
</DbCommand>  
</SelectCommand>  
</DbSource>  
</Sources>  
</TableAdapter>  
</Tables>  
<Sources>  
</Sources>  
</DataSource>  
</xs:appinfo>  
</xs:annotation>  
<xs:element name="ItregMailDetailsTypedDataSet" msdata:IsDataSet="true  
msdata:UseCurrentLocale="true
```

Re: Typed DataSets in VS2005

```
msprop:Generator UserDSName="ItregMailDetailsTypedDataSet"  
msprop:Generator DataSetName="ItregMailDetailsTypedDataSet">  
<xs:complexType>  
<xs:choice minOccurs="0" maxOccurs="unbounded">  
<xs:element name="atablename"  
msprop:Generator UserTableName="atablename"  
msprop:Generator RowDeletedName="atablenameRowDeleted"  
msprop:Generator RowChangedName="atablenameRowChanged"  
msprop:Generator RowClassName="atablenameRow"  
msprop:Generator RowChangingName="atablenameRowChanging"  
msprop:Generator RowEvArgName="atablenameRowChangeEvent"  
msprop:Generator RowEventHandlerName="atablenameRowChangeEventHandler"  
msprop:Generator TableClassName="atablenameDataTable"  
msprop:Generator TableVarName="tableatablename"  
msprop:Generator RowDeletingName="atablenameRowDeleting"  
msprop:Generator TablePropName="atablename">  
<xs:complexType>  
<xs:sequence>  
<xs:element name="mailbox"  
msprop:Generator UserColumnName="mailbox"  
msprop:Generator ColumnVarNameInTable="columnmailbox"  
msprop:Generator ColumnPropNameInRow="mailbox"  
msprop:Generator ColumnPropNameInTable="mailboxColumn">  
<xs:simpleType>  
<xs:restriction base="xs:string">  
<xs:maxLength value="15" />  
</xs:restriction>  
</xs:simpleType>  
</xs:element>  
<xs:element name="storage_group"  
msprop:Generator UserColumnName="storage_group"  
msprop:Generator ColumnVarNameInTable="columnstorage_group"  
msprop:Generator ColumnPropNameInRow="storage_group"  
msprop:Generator ColumnPropNameInTable="storage_groupColumn">  
<xs:simpleType>  
<xs:restriction base="xs:string">  
<xs:maxLength value="10" />  
</xs:restriction>  
</xs:simpleType>  
</xs:element>  
<xs:element name="server"  
msprop:Generator UserColumnName="server"  
msprop:Generator ColumnVarNameInTable="columnserver"  
msprop:Generator ColumnPropNameInRow="server"  
msprop:Generator ColumnPropNameInTable="serverColumn">  
<xs:simpleType>  
<xs:restriction base="xs:string">  
<xs:maxLength value="15" />  
</xs:restriction>  
</xs:simpleType>  
</xs:element>
```

Re: Typed DataSets in VS2005

```
</xs:sequence>  
</xs:complexType>  
</xs:element>  
</xs:choice>  
</xs:complexType>  
<xs:unique name="Constraint1">  
<xs:selector xpath="//mstns:atablename" />  
<xs:field xpath="mstns:mailbox" />  
</xs:unique>  
</xs:element>  
</xs:schema>
```

"Peter Bradley" <pbradley@xxxxxxxxxx> wrote in message
news:ePvmAEgcHHA.4260@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

Nirosh. That's what we've done. That's why I'm asking the
question.

Peter

"Champika Nirosh" <test@xxxxxx> wrote in message
news:ubhnSBgcHHA.4260@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

I am pretty sure that you are doing some
thing wrong here.. can you just
create a seperate type dataset and confirm
that it add a connection
string too?? Why it need to have a
connection string hradcoded to create
a typed dataset for you.. but it you select the
optipon of autocreating
the DAL then yes it does add a hardcoded
connection string to the
dataset..

Nirosh.

"Peter Bradley" <pbradley@xxxxxxxxxx>
wrote in message
news:uHJh0kfcHHA.3644@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

Thanks Nirosh, but I'm not
sure I understood all that.

I'm only talking about a
typed DataSet: not about the
entire data
access layer – which we
code ourselves and which
uses many typed

Re: Typed DataSets in VS2005

datasets. At least that was
the case until now. If typed
datasets
now have the connection
string hard coded into them,
we'll definitely
not be using them any more
– for security reasons apart
from anything
else.

Peter

"Champika Nirosh"
<test@xxxxxx> wrote in
message
news:eLAvOUfcHHA.3960@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

VSS 2005
support auto
generating
the
dataaccess
layer .. and
in that
case you are
free to
change the
code where
it hardcode
the
connection
string so
that it will
read it from
a config file
(obviously
you have
to change
the code
again if you
have to
create/ edit
the dataset
again.

In my view
point the
automatically
generated

Re: Typed DataSets in VS2005

code is very
good if you
are heading
toward a
quick
concept
demo or a
working
prototype..
but
if you are
looking to
develop a
real
extensible
large-scale
system.
then I rather
advise you
to have your
own data
access
layer.
indeed
you may
have some
assistance
from the
Data access
Application
Blocks
or
enterprise
library.
There you
will get that
chance of
developing
a
flexible
dataaccess
layer.. I am
telling this
with my
experience
and
if you go
with the
auto
generated
code you

Re: Typed DataSets in VS2005

Re: Typed DataSets in VS2005

will at times
drive in to
bottleneck
of your
code..

Nirosh.

"Peter
Bradley"
<pbradley@xxxxxxxxxx>
wrote in
message
news:eLioYLfcHHA.2404@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

Hi
all.

This
post
is
sort
of
tangentially
related
to
my
earlier
posts
on
configuration
files
for
DLLs.

Does
anyone
know
how
to
create
typed
DataSets
using
VS2005's
new
DataSet
designer.
but
with

Re: Typed DataSets in VS2005

the
ability
to
configure
the
connection
string
via
a
config
file?
The
designer
seems
to
hard-code
the
connection
string
into
the
dataset
itself,
which
just
can't
be
right.

The
typed
DataSets
created
by
VS2005
comes
with
a
dll.config
file
that
looks
as
though
it's
trying
to
do
something
like
that.

Re: Typed DataSets in VS2005

and
which
is
put
with
the
executable
at
compile
time.
but
altering
the
connection
string
in
there
doesn't
appear
to
have
any
effect.
We've
seen
lots
of
people
asking
this
question.
but
haven't
found
anyone
answering
it
yet.

Thanks
in
advance

Peter

