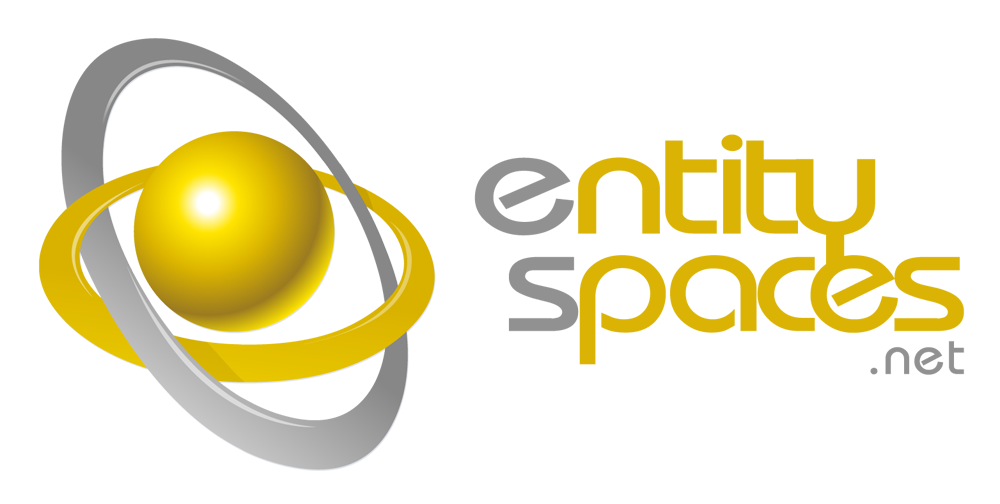
EntitySpaces Miscellaneous Release Notes

Release Notes for Maintenance Releases

August 16, 2011  
EntitySpaces, LLC  
The EntitySpaces Team

**[](http://www.entityspaces.net/)**

Persistence Layer and Business Objects for Microsoft .NET

Contents

[EntitySpaces 2011.1.1110.0 1](#_Toc317028199)

[EntitySpaces 2011.1.0627.0 2](#_Toc317028200)

[EntitySpaces 2011.1.0530.0 3](#_Toc317028201)

[EntitySpaces 2011.1.0516.0 4](#_Toc317028202)

[EntitySpaces 2010 5](#_Toc317028203)

[EntitySpaces 2009.2.1012.0 6](#_Toc317028204)

[EntitySpaces 2009.2.1005.0 7](#_Toc317028205)

[VistaDB 4.0 Support 7](#_Toc317028206)

[Fixed the “||” Issue 7](#_Toc317028207)

[VB.NET Nullable Type Query Issue 7](#_Toc317028208)

[EntitySpaces Alpha 2009.1.0803.0 Alpha 8](#_Toc317028209)

[Sybase Support 8](#_Toc317028210)

[New Video, "Getting Started" PDF, and Samples 8](#_Toc317028211)

[Other Features 9](#_Toc317028212)

[EntitySpaces 2009.1.0706.0 Alpha 10](#_Toc317028213)

[EntitySpaces 2009.1.0629.0 Alpha 11](#_Toc317028214)

[Release Notes 12](#_Toc317028215)

[EntitySpaces 2009.0.1124.0 Alpha 14](#_Toc317028216)

[EntitySpaces 2008.1.1110.0 15](#_Toc317028217)

[EntitySpaces 2008.1.0927.0 16](#_Toc317028218)

[EntitySpaces 2008.1.0922.1 17](#_Toc317028219)

[EntitySpaces 2008.1. 0922.0 18](#_Toc317028220)

[EntitySpaces 2008.1.0811.0 19](#_Toc317028221)

[EntitySpaces 2008.1.0623.0 20](#_Toc317028222)

[New Features and Bug Fixes 20](#_Toc317028223)

[Assemblies 21](#_Toc317028224)

[EntitySpaces 2007.1.1210.0 22](#_Toc317028225)

[EntitySpaces 2007.1.1119.0 23](#_Toc317028226)

[EntitySpaces 2007.1.1024.0 Beta 24](#_Toc317028227)

[EntitySpaces 2007.1.1021.0 Beta 25](#_Toc317028228)

# EntitySpaces 2011.1.1110.0

<http://www.entityspaces.net/blog/2011/11/11/EntitySpaces+Version+2011111100+Released.aspx>

# EntitySpaces 2011.1.0627.0

A minor maintenance release:

Fixed the DataGridView sorting issue when columns names do not match property names.

Fixed the Design-time binding issue with hierarchical properties.

# EntitySpaces 2011.1.0530.0

A minor maintenance release.

<http://www.entityspaces.net/blog/2011/05/31/EntitySpaces+2011105300+Released.aspx>

# EntitySpaces 2011.1.0516.0

<http://www.entityspaces.net/blog/2011/05/16/Update+2011105160+Released.aspx>

# EntitySpaces 2010.1.1122.0

<http://www.entityspaces.net/blog/2010/11/22/EntitySpaces+Maintenance+Release+2010111220+Released.aspx>

# EntitySpaces 2010.1.1025.0

<http://www.entityspaces.net/blog/2010/10/25/EntitySpaces+2010110250+Released+To+Production.aspx>

# EntitySpaces 2010.1.0830.0

EntitySpaces Adds Support for SQL Azure and SQL Server Compact 4.0.

<http://www.entityspaces.net/blog/2010/08/30/EntitySpaces+Adds+Support+For+SQL+Azure+And+SQL+Server+Compact+40.aspx>

# EntitySpaces 2010.1.0816.0

This release is recommended if your application is a Windows.Forms application. We fixed a bug that happened when the user pressed the “escape” key while editing in a grid. If you are upgrading from our 2010.1.0802.0 release there is no need to regenerate your classes.

# EntitySpaces 2010.1.0802.0

<http://www.entityspaces.net/blog/2010/08/01/EntitySpaces+2010108020+Available+For+Download.aspx>

# EntitySpaces 2010.1.0726.0

This is a minor release. There is no need to upgrade unless you need one of the fixes below. The fixes are as follows:

* The WCF Client Side Proxy Patch is included (it was not included with 2010.1.0720.0)
* The namespace "Proxies" was hard coded in the client proxy template, this was fixed.
* The “char” modified column count error has been fixed.
* The case sensitive bug when returning columns with a different case than defined in the actual table error has been fixed.

# EntitySpaces 2010.1.0720.0

This is basically a maintenance release. For most folks doing EntitySpaces 2010 development, we do recommend upgrading. No code generation is required unless one of the issues mentioned below fixes an issue you were having.

Issues this maintenance release addresses:

* The WCF Templates we’re tweaked. There was an case where we accidentally used the “PropertyName” instead of the “ColumnName”.
* If your application uses the “Connection Name” feature on the “Generated Master” template this issue has been fixed (just regenerate).
* The Set accessor for columns of type “Byte[]” were failing if they were the first property to be set on an object, this has been fixed.
* There was a case that would cause a stack overflow in VB.NET if the ForEach syntax was used in a “Shared” method (very obscure but fixed).
* We upgraded to the new SandleCastle help file system. The help files are quite a bit different in appearance now.

# EntitySpaces 2010.1.0706.0

<http://www.entityspaces.net/blog/2010/07/06/EntitySpaces+2010107060+Available+For+Download.aspx>

# EntitySpaces 2009.2.1214.0

<http://www.entityspaces.net/blog/2009/12/12/EntitySpaces+200921214+Released.aspx>

# EntitySpaces 2009.2.1012.0

This is a minor maintenance release to fix an issue for SQL Server when a table has both a default NEWSEQUENTAILID() primary key and a timestamp column.

# EntitySpaces 2009.2.1005.0

This is a maintenance release to support VistaDB 4.x and fix a couple of VB.NET issues.

## VistaDB 4.0 Support

We have made a new version of EntitySpaces Developer available (2009.2.1005.0) that supports VistaDB 4.0. The official release of VistaDB 4.0 was just made available and within 24 hours of its release we have added support for it. The EntitySpaces Developer version is available but we haven’t updated the Trial version yet. That will happen within the next 24 hours or so. You can install this right over the top of the 2009.2.0928.0 release. Make sure you have Visual Studio closed. There are now two choices on the Settings tab when connecting to VistaDB.

VistaDB = VistaDB 3.x   
VistaDB4 = VistaDB 4.x

Also, there is a new EntitySpaces DataProvider that is used with VistaDB 4.x. The assembly name is EntitySpaces.VistaDB4Provider.dll. In your config file in the EntitySpaces connection information you use “EntitySpaces.VistaDB4Provider” as the provider. There are no VistaDB 4.0 compact framework providers as VistaDB 4.0 does not currently ship a Compact Framework assembly.

## Fixed the “||” Issue

There was an issue with the || (or) symbol. This has been fixed. In C# you should always be able to use && for AND and || for OR. Of course, in VB.NET you just use the OR and AND keywords.

## VB.NET Nullable Type Query Issue

There was an issue that was causing VB.NET developers to use .Value when using nullable types in DynamicQuery statements. This has been fixed.

# EntitySpaces 2008.1.1110.0

There were two bugs involving the deleting of data that have been fixed:

The first bug happened when you did a MarkAllAsDeleted() and then called Save, it wasn't actually deleting all of the records. This has been fixed and was introduced last release when we made the performance tweaks.

The 2nd bug involved deleting data in a hierarchical scenario and we traced this back all the way to ES2007 so it had been there awhile, however it is also fixed now. The scenario that was failing is shown below.

OrderCollection collection **=** **new** OrderCollection**();**

collection**.**LoadAll**();**

Order ord **=** collection**.**FindByPrimaryKey**(**10**);**

ord**.**OrderItemCollectionByOrderID**.**MarkAllAsDeleted**();**

ord**.**MarkAsDeleted**();**

collection**.**Save**();**

# EntitySpaces 2008.1.0927.0

<http://www.entityspaces.net/blog/2008/09/29/EntitySpaces+2008+0927+Released.aspx>

# EntitySpaces 2008.1.0922.1

Okay, this should be it. There was an issue with adding new items through a grid in windows forms editing. This has been fixed in 0922.1 (which is different than 0922.0) and we recommend that everyone just upgrade. It is not necessary to regenerate your code if you already have with 0915 and up.

# EntitySpaces 2008.1.0922.0

There were a few bugs found in our 0915 release and we felt the need to put out a new release to address them. We recommend that everyone on the 0915 release upgrade to our 0922 release. If you regenerated your classes on the 0915 release there is no need to regenerate your classes again on the 0922 release. We will post a 0922 Trial version very soon. You will know when the new Trial version is available by looking under the download button on our home page.

Here are the issues that were addressed.

There was a bug when binding to a grid and adding a new row through the grid interface. This has been fixed.

There was a bug in certain complex hierarchical save scenarios. This has also been fixed.

We have a slew of improvements and requests that we are going to pour into ES2009. We have already converted our entire suite of C# templates to run under our new code generator so things are moving forward very nicely on ES2009.

# EntitySpaces 2008.1.0811.0

EntitySpaces 2008 was originally released on June 23rd and it has proven to be very solid, as have our past releases. We decided to put out a maintenance release to make some minor tweaks we have made available. We have also upgraded the ASPX Admin Suite to run under ES2008. The ASPX Admin suite is only available under MyGeneration as the UI technique it uses is not compatible with CodeSmith. You can install this release right over the top of your current ES2008 installation.

Upgraded our C# and VB.NET Web ASPX Admin Grid templates for ES2008 (MyGeneration Only).

Fixed Oracle SEQ double-quotes in MyGeneration VB Metadata and MetadataMap templates.

Add support for Oracle FLOAT data type.

We no longer use the deprecated Oracle Precision and Scale in parameters for NUMERIC data types.

Fixed a casting issue for esDataSource TotalRowCount with MySQL and Oracle.

Add Property Changed Notification to all four hierarchical properties.

Fixed exception handling in FillDataTable and FillDataSet for all providers.

Updated template UI and "Getting Started" PDFs to state framework requirements for LinqToSql(3.5) and WCF(3.0) options.

# EntitySpaces 2008.1.0623.0

Before we begin we want to let you know that you should uninstall any prior ES2008 beta's including the release candidate before installing the official release. The official release is version number 2008.1.0623.0 (yyyy.major.mmdd.minor) and passes all of our regressions tests. We were also able to sneak in fixes and enhancements such as support for the new "rowversion" data type in Microsoft SQL Server CE which provides support for concurrency checking, and other fixes as well. The installer also now installs two versions of our .CHM based help system. We still ship the full blown help system but we also provide a new scaled down "simple" version which you will find on your menu after installing. This simple version should prove very useful if you are just getting started with EntitySpaces.

## New Features and Bug Fixes

* Support for both MyGeneration and CodeSmith.
* LINQ Support for our SQL Server and SQL Server CE Desktop providers.
* EntitySpaces 2008 - [LINQ to SQL in Next Beta](http://www.entityspaces.net/blog/2008/05/11/EntitySpaces+2008+LINQ+To+SQL+In+Next+Beta.aspx)
* EntitySpaces 2008 - [LINQ to SQL in Next Beta (Part 2)](http://www.entityspaces.net/blog/2008/05/11/EntitySpaces+2008+LINQ+To+SQL+In+Next+Beta+Part+2.aspx)
* Added "[Casting](http://www.entityspaces.net/blog/2008/03/22/EntitySpaces+2008+QuotCastquot+Support.aspx)" Support
* Added Subquery Support
* EntitySpaces 2008 - [Dynamic SubQuery Showcase](http://www.entityspaces.net/blog/2008/03/03/EntitySpaces+2008+Dynamic+SubQuery+Showcase.aspx)
* EntitySpaces 2008 [Adds Subqueries](http://www.entityspaces.net/blog/2008/02/08/EntitySpaces+2008+Adds+Subqueries.aspx)
* Support for the SQL Server 2005 NEWSEQUENTIALID() Function.
* Support for the new SQL Server 2008 Datatypes.
* Added Query.SelectAllExcept() - Great for excluding large blob type columns when you don't really need them.
* Added PostgreSQL Guid support (see EntitySpaces.Npgsql2Provider.dll below).
* Fixed obj.Query.Where(new esWhereItem(esParenthesis.Close)) adds extra default Conjunction.
* Added the ability to un-hook a Query from a collection.
* Leaving esDataSource.TotalRowCount set to -1 it will cause it to calculate it for you (no need for a second query).
* Better exception handling. All low level database exceptions should bubble up to the application code. No more "Object reference not set" errors.
* The installer also ships full source for the EntitySpaces.MetadataEngine.dll plugins for both SQL Server CE and VistaDB so you can recompile these if you need to use different versions.
* All assemblies have new strong name keys and the source code does not come with the .SNK files, if you need to recompile the source code you will need to generate your own EntitySpaces.snk files.
* Generating code with "Generate Single File" set to false should no longer have any errors.
* Fixed Query.OP() LIKE/NOT LIKE for public use.
* Fixed parameter direction in Oracle delete stored procedures.
* Fixed property settor when INotify and Hierarchical are both checked
* Properties that are not brought back by queries no longer throw exceptions, they simply return null if accessed. This is very nice for our proxies as you no longer have to return all columns for the proxies to work properly.
* Many other minor fixes and enhancements were made.

## Assemblies

The Standard Assemblies for .NET 2.0 and .NET 3.5

Assembly Name and Description

* EntitySpaces.Core.dll - The EntitySpaces Core classes
* EntitySpaces.Interfaces.dll - The EntitySpaces Interfaces
* EntitySpaces.Loader.dll - Provider Loader - Uses Reflection
* EntitySpaces.LoaderMT.dll - Provider Loader - No Reflection (Medium Trust)
* EntitySpaces.Web.dll - ASP.NET - esDataSource
* EntitySpaces.Web.Design.dll - ASP.NET - esDataSource Design Time Support
* EntitySpaces.MSAccessProvider.dll - Provider - Microsoft Access
* EntitySpaces.MySqlClientProvider.dll - Provider - MySQL Data Provider
* EntitySpaces.Npgsql2Provider.dll - Provider - PostgreSQL (Npgsql 2.x Beta 3, Guid Support)
* EntitySpaces.NpgsqlProvider.dll - Provider - PostgreSQL (Npgsql 1.0)
* EntitySpaces.OracleClientProvider.dll - Provider - Oracle
* EntitySpaces.SqlClientProvider.dll - Provider - Microsoft SQL Server
* EntitySpaces.SqlServerCeProvider.dll - Provider - Microsoft SQL Server CE (Desktop)
* EntitySpaces.VistaDBProvider.dll - Provider - VistaDB

The Compact Framework Assemblies for .NET 2.0 and .NET 3.5

Assembly Name and Description

* EntitySpaces.Core.CF.dll - The EntitySpaces Core classes
* EntitySpaces.Interfaces.CF.dll - The EntitySpaces Interfaces
* EntitySpaces.Loader.CF.dll - Provider Loader - Uses Reflection
* EntitySpaces.LoaderMT.CF.dll - Provider Loader - No Reflection
* EntitySpaces.SqlServerCeProvider.CF.dll - Provider - Microsoft SQL Server CE
* EntitySpaces.VistaDBProvider.CF.dll - Provider - VistaDB

There is a special version of the EntitySpaces.Core.dll in the "Web" subfolder that allows for better binding in ASP.NET scenarios when extra columns are brought back by joins and by other means. This should eliminate the need to use LowLevelBind().

Both our .NET 2.0 and .NET 3.5 EntitySpaces.SqlServerCeProvider.CF.dll's are bound to the 3.5.0.0 version of System.Data.SqlServerCe.dll. If you need to redirect the bindings you can do so. Both our .NET 3.5 versions of EntitySpaces.SqlClientProvider.dll and EntitySpaces.SqlServerCeProvider.dll (Desktop) have support for LINQ.

# EntitySpaces 2007.1.1210.0

Fixed old style mixed nested AND and OR.

Fixed Oracle % operator to use MOD() function.

Fixed Oracle overflow error for some division results, e.g., 1.3333.... with ROUND(expr, 10) (same as STDDEV and VAR).

Fixed Oracle Paging with Joins.

Fixed MySQL CONCAT() expression.

Fixed configless esConfigSettings.ConnectionInfo.Default settor for EntitySpaces.Interfaces.Ce.

Added new class called esEntityCollectionView. You can create many of these on a single esEntityCollection and sort/filter them all separately. See this blog post for details.

Updated XML comments in VB and C# entity and collection templates for generated classes.

Fixed delimiter issue for SqlCe when concatenating (+ operator) string columns with literals.

Added overload to Like/NotLike operators that takes an ESCAPE character, e.g., where Northwind Products QuantityPerUnit does not have "10%" in it. The "%" needs to be escaped, so it does not behave like a wild card character.

# EntitySpaces 2007.1.1119.0

New Client Side Proxies Available, see this BLOG post.

Fix an error that would occur after restart, see this forum POST

All ProxyStubs are now partial classes, some common logic that was in every ProxyStub class was pushed into a base class. There is now a GetEntityType() method on the proxies that can be used to get the esEntity's Type.

You can now run without defining a default connection, and assign connections as needed. This is great for hosted web applications that have different databases per client.

New DynamicQuery feature - The "Contains" Where operator is implemented for SQL Server (only). It requires that Full Text indexing be enabled. See this FAQ.

On the "Generated Master" template both "Metadata Class Should Ignore Schema" and "Metadata Class Should Ignore Catalog" are now checked by default.

# EntitySpaces 2007.1.1024.0 Beta

ASP.NET Databinding has been fixed

The OnPropertyChanged() bug has been fixed

The esDataSource should now automatically bring up all of your collections. Hopefully you will no longer have to browse to the dll.

# EntitySpaces 2007.1.1021.0 Beta

Things changed since our last beta. We no longer us any attributes for our hierachical binding. Our Compact Framework support prevents us from using attributes as we do not intend on supporting two different codebases. Thus, our databinding was reworked from our last beta and it is so much easier, and better. This release fully supports the Compact Framework. We really encouage you to use this version if you can. If you are doing Windows.Forms development you will really enjoy the power in this build.

We feel this is a very solid beta, our Compact Framework support works just fine as well.

Windows.Forms Hierarchical Databinding

For Windows.Forms developers hierarchical binding is now supported. There is new flag on your collection classes called EnableHierarchicalBinding that is true by default. You can turn if off in the designer by selecting your collection and setting that property to false.

Extended Properties

The old method CreateExtendedProperties() is no longer supported and no longer exists. The concept of extended properties themselves really no longer exist. They are simply not needed. You will see as you read further.

Virtual Columns

A virtual column is any column that is brought back from a query, a stored procedure, or any other data fetching mechanism that is not in your core table or view. For instance, if you use our new "join" syntax to bring back any extra columns from other tables, these extra columns are referred to as virtual columns. All virtual columns are now bindable at runtime by default. For instance, if you bring back an extra column named "FullName" that column will exist as a "virtual property" for binding purposes only, and can be seen automatically by whatever you are binding to (not via intellisense though).

Local Binding Properties

Local binding properties always show up in design time mode. There are two classes, both shown below. All columns added during GetLocalBindingProperties() will always be available in design time.

The FakeColumn property below must be added in the GetLocalBindingProperties() if you want to make it available during runtime. The reason you must do this is that it is not contained in the underlying DataTable and EntitySpaces knows nothing about it. Of course, adding it here also means it will be available in design time.

The FullName property below is only added to the GetLocalBindingProperties() because we want it to show up in design time. Since it is returned in the underlying DataTable it will aways be available at runtime.

**namespace** BusinessObjects

**{**

**public** partial class Employee **:** esEmployee

**{**

**protected** **override** List GetLocalBindingProperties**()**

**{**

List props **=** **new** List**();**

props**.**Add**(new** esPropertyDescriptor**(this,** "FakeColumn"**,** **typeof(**string**)));**

props**.**Add**(new** esPropertyDescriptor**(this,** "FullName"**,** **typeof(**string**)));**

**return** props**;**

**}**

**public** string FakeColumn

**{**

get **{** **return** f**;** **}**

set **{** f **=** value**;** **}**

**}**

**public** string FullName

**{**

get **{** **return** **(**string**)this.**GetColumn**(**"FullName"**);** **}**

set **{** **this.**SetColumn**(**"FullName"**,** value**);** **}**

**}**

**private** string f **=** "wow man"**;**

**}**

**}**

IDataErrorInfo Implementation

The IDataErrorInfo Implementation lives in the esEntity.cs class (your single entities base class) and here is the actual implementation.

#region IDataErrorInfo

Membersstring IDataErrorInfo**.**Error

**{**

get

**{**

**return** **((**IDataErrorInfo**)this)[**string**.**Empty**];**

**}**

**}**

string IDataErrorInfo**.this[**string columnName**]**

**{**

get

**{**

string result **=** string**.**Empty**;**

esColumnMetadata columnMeta **=** **this.**Meta**.**Columns**[**columnName**];**

**if** **(this.**OnValidateDelegate **!=** **null)**

**{**

result **=** OnValidateDelegate**(**columnName**,** **this,** columnMeta**);**

**}**

**if** **(**string**.**IsNullOrEmpty**(**result**))**

**{**

result **=** **this.**Validate**(**columnName**,** **this,** columnMeta**);**

**}**

**return** result**;**

**}**

**}**

**virtual** **public** string Validate**(**string columnName**,** esEntity entity**,** esColumnMetadata metadata**)**

**{**

**return** string**.**Empty**;**

**}**

**public** **delegate** string ValidateDelegate**(**string columnName**,** esEntity entity**,** esColumnMetadata metadata**);**

**public** **event** ValidateDelegate OnValidateDelegate**;**

#endregion

There are two ways to use take advantage of the IDataErrorInfo implementation, by delegate, or by a virtual/overrideable method. Below are a few examples. They might not make to much sense, but they should show you the simple mechanics. The nice thing about this approach is that you are handed the Column Name, the actual esEntity, and the esColumnMetadata for the column.

To use the virtual Validate() method you simply overload/Override it in your custom class like this:

**namespace** BusinessObjects

**{**

**public** partial class Employees **:** esEmployees

**{**

**public** **override** string Validate**(**string columnName**,** esEntity entity**,** esColumnMetadata metadata**)**

**{**

Employees employeeToValidate **=** entity **as** Employees**;**

**if** **(**employeeToValidate **!=** **null)**

**{**

**if** **(**columnName **==** EmployeesMetadata**.**ColumnNames**.**LastName **&&**

employeeToValidate**.**LastName**.**Length **<** 15**)**

**{**

**return** "Virtual method says we do not like employees with short last names"**;**

**}**

**}**

**return** string**.**Empty**;**

**}**

**}**

**}**

To use the delegate you would take this approach in your winform (or wherever)

**private** void FirstNameTextBox\_Validated**(object** sender**,** EventArgs e**)**

**{**

emp**.**OnValidateDelegate **=** **new** EntitySpaces**.**Core**.**esEntity**.**ValidateDelegate**(**emp\_OnValidateDelegate**);**

Control control **=** **(**Control**)**sender**;**

**foreach** **(**Binding binding **in** control**.**DataBindings**)**

**{**

**if** **(**binding**.**IsBinding**)**

**{**

IDataErrorInfo errorInfo **=** **(**IDataErrorInfo**)**binding**.**DataSource**;**

errorProvider1**.**SetError**(**control**,** errorInfo**[**binding**.**BindingMemberInfo**.**BindingField**]);**

**}**

**}**

**}**

string emp\_OnValidateDelegate**(**string columnName**,** EntitySpaces**.**Core**.**esEntity entity**,**

EntitySpaces**.**Interfaces**.**esColumnMetadata metadata**)**

**{**

Employees employeeToValidate **=** entity **as** Employees**;**

**if** **(**employeeToValidate **!=** **null)**

**{**

**if** **(**columnName **==** EmployeesMetadata**.**ColumnNames**.**FirstName **&&**

employeeToValidate**.**FirstName**.**Length **<** 15**)**

**{**

**return** "The delegate says we do not like people with short first names"**;**

**}**

**}**

**return** string**.**Empty**;**

**}**

Hierarchical Properties - Forcing a Reload (breaking change)

All hierarchical properties now have a "set" which you can call. However, you can only set it to Null / Nothing otherwise an exception will be thrown. The set looks something like this:

set

**{**

**if** **(**value **!=** **null)**

**throw** **new** Exception**(**"'value' Must be null"**);**

**if** **(this.**\_EmployeeCollectionBySupervisor **!=** **null)**

**{**

**this.**RemovePostSave**(**"EmployeeCollectionBySupervisor"**);**

**this.**\_EmployeeCollectionBySupervisor **=** **null;**

**}**

**}**

This means you can now do this…

employee.EmployeeCollectionBySupervisor[1].FirstName = "mike"; // lazy loaded upon first access employee.EmployeeCollectionBySupervisor = null; // discard it employee.EmployeeCollectionBySupervisor[1].FirstName = "mike"; // the entire collection was re-loaded

The breaking change comes into play for esEntity based hierarchical objects, you use to be able to perform this operation.

employee.UpToEmployeeBySupervisor = employeeObject;

now you must do:

employee.Supervisor = employeeObject.EmployeeID;

Nullable Types - Experimental (at this point we suggest you don't use this)

If you run the EntitySpaces - Set esPlugin Settings template you will see a new checkbox named "Use Nullable Types Always" which is set to true. This is how we have always handled nullable types. However, if you set this to "false" then only columns that allow null will be nullable types. We have done some testing on this, however, for those of you that are interested in this support, we need your help in testing. The only drawback right now is that our proxy's do not support this feature yet, so if you check the proxy checkbox and generate the proxies you will get compile errors.

SQL Server Paging with Joins

This bug was fixed.

VistaDB 3.2 - Build 43 (it was still warm, you could just smell the butter)

We are now compiled against VistaDB 3.2 Build 43.

\*\* Special PostgreSQL instructions \*\*

myGeneration ships with the PostgreSQL native column type to C# mappings named differently than the EntitySpaces templates expect them. However, this can be easily corrected in a few quick seconds by following these these steps (VB.NET users follow these same steps, EntitySpaces uses only the C# mappings even for VB.NET):

Open MyGeneration

Connect to your PostgreSQL database using the MyGeneration Default Settings dialog. It's under the "Edit -> Default Settings ..." menu

Click on the "Language Mappings" toolbar button (4th from the right - use tooltips)

Select "C# System Types" in the drop down combobox in the "Language Mappings" dockable window.

Click on the "Create New Language Mapping" icon (white icon on the "Language Mappings" dockable window)

In the popup dialog for the Language type in C# and Based Upon select C# System Types

Click on the "OK" button

Click on the "Save" icon on the "Language Mappings" dockable window

Close and reopen MyGeneration, you're all set.

The above instructions are a one time operation, you will not have to do them again.

The EntitySpaces.NpgsqlProvider.dll uses Npgsql version 1.0. It should be installed with your PostgreSQL 8.2 installation.

# EntitySpaces 2007

EntitySpaces 2007 introduced the hierarchical model.

# EntitySpaces 2006 V1

EntitySpaces first commercial offering began shipping in February, 2006. It was a top-to-bottom re-write of the popular, free dOOdads architecture for Microsoft .NET 1.0. It was re-written to take advantage of the new features in Microsoft .NET 2.0.