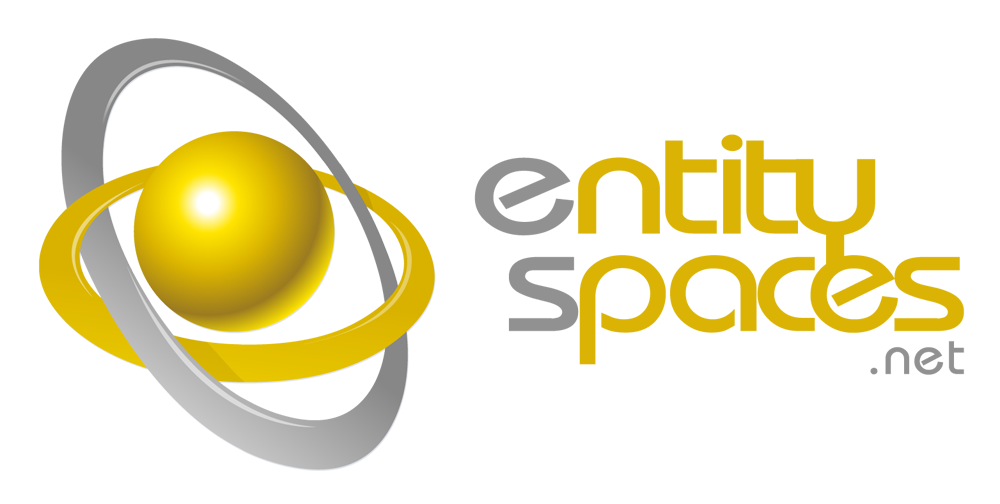
EntitySpaces 2010 Release Notes

June 28th, 2010

EntitySpaces, LLC

The EntitySpaces Team

**[](http://www.entityspaces.net/)**  
Persistence Layer and Business Objects for Microsoft .NET

Contents

[Requirements 3](#_Toc265336229)

[Microsoft .NET 3.5 Framework 3](#_Toc265336230)

[Installation 3](#_Toc265336231)

[Breaking Changes 4](#_Toc265336232)

[Custom Classes 4](#_Toc265336233)

[esEntity.AddNew() 5](#_Toc265336234)

[esDataRowState 5](#_Toc265336235)

[esEntity.GetColumn 5](#_Toc265336236)

[Query.QueryReset() 5](#_Toc265336237)

[esEntityCollection.RowStateFilter 5](#_Toc265336238)

[esEntityCollection.Sort 5](#_Toc265336239)

[esEntityCollection.Filter 5](#_Toc265336240)

[esEntityCollection.LowLevelBind() 6](#_Toc265336241)

[SQL Server mappings added for "char" data type 6](#_Toc265336242)

[Custom Base Classes 6](#_Toc265336243)

[Project Files are Now Path Relative 6](#_Toc265336244)

[Non Breaking Changes 7](#_Toc265336245)

[Generated Collection Classes 7](#_Toc265336246)

[EmployeesCollection 7](#_Toc265336247)

[esEmployeesCollection 7](#_Toc265336248)

[esEntityCollection 8](#_Toc265336249)

[Generated Entity Classes 9](#_Toc265336250)

[Employees 9](#_Toc265336251)

[esEmployees 9](#_Toc265336252)

[esEntity 9](#_Toc265336253)

[SQLite Support 10](#_Toc265336254)

[Known Issues 10](#_Toc265336255)

[Blog Posts on EntitySpaces 2010 10](#_Toc265336256)

# Requirements

## Microsoft .NET 3.5 Framework

The EntitySpaces 2010 architecture requires the Microsoft .NET 3.5 Framework. It does not support running under .NET 2.0. Customers that wish to develop for the .NET 2.0 Framework will have to remain on EntitySpaces 2009.

# Installation

EntitySpaces 2010 can be installed side-by-side with EntitySpaces 2009. You cannot install two copies of EntitySpaces 2010 side-by-side. We recommend taking the defaults during installation. For Vista and Win7 installations, you should right-click the installer and "Run as Administrator". There is a "Manual Setup" PDF installed to your Start Menu if you have issues using the EntitySpaces 2010 AddIn in Visual Studio.

# Breaking Changes

## Custom Classes

The ability to place custom constructors in the Custom classes has been an often-requested feature. When custom classes are generated in EntitySpaces 2010, they have constructors in them and all of the necessary namespaces needed for adding custom methods. The EntitySpaces Custom Master template will not overwrite existing files since the Custom classes contain your custom business logic.

For new projects using EntitySpaces 2010, you must generate both the Custom Master and Generated Master templates.

For upgrading projects from EntitySpaces 2009 to EntitySpaces 2010, there are a few approaches you can take to address migrating the Custom classes.

1. You can manually add the constructors to your existing Custom classes.
2. You can generate the EntitySpaces 2010 Custom classes and copy/paste your existing custom code into them.
3. For projects that have many Custom classes, you can take advantage of the fact that they are generated as partial classes.
   1. Create a new folder below Custom called CustomConstructors
   2. Generate the ES2010 Custom Master classes into the new CustomConstructors folder

Generating the Custom Master template, or migrating your Custom classes need only be done once.

*ES2009*

using EntitySpaces.Interfaces;  
using EntitySpaces.Core;

namespace BusinessObjects  
{  
 public partial class Categories : esCategories  
 {

}  
}

*ES2010*

using EntitySpaces.Core;  
using EntitySpaces.Interfaces;  
using EntitySpaces.DynamicQuery;  
  
namespace BusinessObjects  
{  
 public partial class Categories : esCategories  
 {  
 **public Categories()  
 {  
  
 }**  
 }  
}

## esEntity.AddNew()

This method has been removed. It was optional in 2009 and has finally been removed. If you have overloaded or overridden this virtual method in a Custom class, you should move that logic to the constructor in your Custom classes (see above) and remove the custom AddNew method.

## esDataRowState

Since we no longer use a DataRow to hold an entity's data, we no longer use the ADO.NET **DataRowState** enum. Instead, we use our own enum esDataRowState. Single entities have an esDataRowState of “Added” upon creation and IsDirty is true.

## esEntity.GetColumn

This method now returns null or Nothing and not DBNull.Value.

## Query.QueryReset()

This method no longer exists, you never re-use existing queries, you simply instantiate new ones.

## esEntityCollection.RowStateFilter

This property no longer exists. To do RowStateFiltering in EntitySpaces 2010:

coll.Filter = coll.AsQueryable().Where  
(  
 d => d.es.RowState == esDataRowState.Modified ||  
 d.es.RowState == esDataRowState.Unchanged  
);

You cannot filter deleted entities, but you can iterate over deleted entities in the **coll.es.DeletedEntities** property.

## esEntityCollection.Sort

This property no longer exists. You now use the esEntityCollection.Filter property to sort, filter, or perform both operations at the same time. See below.

## esEntityCollection.Filter

This property is no longer a “string” type. Instead, it is of type IQueryable<T>. This method no longer relies on ADO.NET’s quirky syntax and instead uses the new Microsoft .NET 3.5 syntax. Note: you must have a **using System.Linq** statement in order to see the intellisense for AsQueryable. See the following examples:

* *Sorting by the Employee’s first name*

coll.Filter = coll.AsQueryable().OrderByDescending(d => d.FirstName);

* *Filtering for all Empoyees missing their FirstName*

coll.Filter = coll.AsQueryable().Where(d => d.FirstName == null);

* *Both Filtering and Sorting*

coll.Filter = coll.AsQueryable().Where(d => d.FirstName == null).OrderByDescending(d => d.LastName);

## esEntityCollection.LowLevelBind()

This method no longer exists. It is no longer necessary.

## SQL Server mappings added for "char" data type

Mappings were added to esLanguages.xml to map SQL Server char(1), nchar(1), varchar(1), and nvarchar(1) data types to "char". In code, this will generate properties with System.Char? by default for those mappings rather than System.String.

You can go through your compile errors and fix the issues you encounter E.g., C# char literals should be delimited with single quotes rather than the double quotes used for string literals. For VB.NET you place a lower case "c" after the closing double-quote delimiter.

entity.MyCharProperty = "/"**c**

Alternatively, the EntitySpaces Mappings tab is now editable. If you wish to revert to EntitySpaces 2009 behavior, you can delete those four entries or change their mappings to "string", click the Save icon, and re-generate.

## Custom Base Classes

The shipping Custom Base Classes for EntitySpaces 2010 have changed slightly to reflect the changes in the architecture. If you are using the optional, advanced “Custom Base Class” feature of EntitySpaces, and have custom code in those files, then you should manually replicate the changes to them in your project. You will find them in the **C:\Program Files\EntitySpaces 2010\CodeGeneration\CustomBase** folder (just as in EntitySpaces 2009). If you have not added custom code, then you can just copy the new versions over the old.

## Project Files are Now Path Relative

The EntitySpaces 2010 Project Files are path relative. To share them in a team environment check them into your source control system along with the esUserData.xml file if you use Alias or save the user metadata for any other reason.

# Non Breaking Changes

## Generated Collection Classes

The Collection classes have changed the most. However, this should not really affect your current code base. We preserved the API as much as possible.

### EmployeesCollection

* ES2009

public partial class **EmployeesCollection** : esEmployeesCollection

* ES2010

public partial class **EmployeesCollection** : esEmployeesCollection, IEnumerable<Employees>

### esEmployeesCollection

* ES2009

abstract public class **esEmployeesCollection** : esEntityCollection

* ES2010

abstract public class **esEmployeesCollection** : esEntityCollection<Employees>

### esEntityCollection

* ES2009

public abstract class **esEntityCollection** : esEntityData,   
 IEntityCollection,   
 IComponent,   
 IDisposable,   
 IBindingList,   
 IList,   
 ICollection,   
 IEnumerable,   
 ITypedList,   
 IListSource,   
 ICommittable

* ES2010

public abstract class **esEntityCollection<T>** : esEntityCollectionBase,   
 ICommittable,   
 IComponent,   
 IDisposable,   
 IList<T>,   
 ICollection<T>,   
 IEnumerable<T>,   
 ITypedList,   
 IBindingList,   
 IList,   
 ICollection,   
 IEnumerable,   
 ICancelAddNew,   
 IRaiseItemChangedEvents   
 where T : EntitySpaces.Core.esEntity, new()

## Generated Entity Classes

### Employees

* ES2009 (no change)

public partial class **Employees** : esEmployees

* ES2010 (no change)

public partial class **Employees** : esEmployees

### esEmployees

* ES2009

abstract public class **esEmployees** : esEntity, INotifyPropertyChanged

* ES2010

abstract public class **esEmployees** : esEntity

### esEntity

* ES2009

public abstract class **esEntity** : esEntityData, IEntity, IEditableObject, IDataErrorInfo, ICommittable

* ES2010

public class **esEntity** : IEditableObject, IEntity, ICommittable, INotifyPropertyChanged, IDataErrorInfo

# SQLite Support

We have added support for SQLite. You should be able to connect and generate your EntitySpaces classes as normal. The provider is named **EntitySpaces.SQLiteProvider.dll** and is bound to the 1.0.66.0 version. The SQLite ADO.NET provider can be found here: <http://sqlite.phxsoftware.com/>

connectionString="Data Source=C:\Databases\Northwind.db3;Version=3;"

# Known Issues

* The IDataErrorInfo implementation on the Collection classes is not complete
* The VB.NET Silverlight Sample has some issues (probably due to porting) but will be fixed upon the next release.
* Our Windows.Forms sample in this release will be completely re-written. We just shipped our test program(s).

# Blog Posts on EntitySpaces 2010

[**EntitySpaces 2010 Sneak Preview (I)**](http://www.entityspaces.net/blog/2009/12/27/EntitySpaces+2010+Sneak+Preview+I.aspx)

[**EntitySpaces 2010 Sneak Preview (II)**](http://www.entityspaces.net/blog/2010/01/04/EntitySpaces+2010+Sneak+Preview+II.aspx)

[**EntitySpaces 2010 Sneak Preview (III)**](http://www.entityspaces.net/blog/2010/01/11/EntitySpaces+2010+Sneak+Preview+III.aspx)

[**EntitySpaces 2010 Sneak Preview (IV)**](http://www.entityspaces.net/blog/2010/01/21/EntitySpaces+2010+Sneak+Preview+IV.aspx)

[**EntitySpaces 2010 Supports the SQL Server 2008 types SqlGeometry, SqlGeography, more …**](http://www.entityspaces.net/blog/2010/01/24/EntitySpaces+2010+Supports+The+SQL+Server+2008+Types+SqlGeometry+SqlGeography+More+Hellip.aspx)

[**EntitySpaces 2010 – More Template Friendly**](http://www.entityspaces.net/blog/2010/01/24/EntitySpaces+2010+Ndash+More+Template+Friendly.aspx)

[**EntitySpaces 2010 SQLite Support**](http://www.entityspaces.net/blog/2010/01/30/EntitySpaces+2010+SQLite+Support.aspx)

[**EntitySpaces 2010 – Killer Silverlight Binding**](http://www.entityspaces.net/blog/2010/02/25/EntitySpaces+2010+Ndash+Killer+Silverlight+Binding.aspx)

[**EntitySpaces 2010 Two Part Series on Silverlight and WCF**](http://www.entityspaces.net/blog/2010/02/27/EntitySpaces+2010+Two+Part+Series+On+Silverlight+And+WCF.aspx)

[**JSON Serialization and EntitySpaces**](http://www.entityspaces.net/blog/2010/03/04/JSON+Serialization+And+EntitySpaces.aspx)

[**EntitySpaces 2010 Better Handling of Schemas**](http://www.entityspaces.net/blog/2010/03/09/EntitySpaces+2010+Better+Handling+Of+Schemas.aspx)

[**EntitySpaces 2010 WCF Service Creating Template**](http://www.entityspaces.net/blog/2010/04/17/EntitySpaces+2010+WCF+Service+Creating+Template.aspx)

[**EntitySpaces 2010 WPF Sample, Client Side Proxies, and ExtraColumns**](http://www.entityspaces.net/blog/2010/05/08/EntitySpaces+2010+WPF+Sample+Client+Side+Proxies+And+ExtraColumns.aspx)