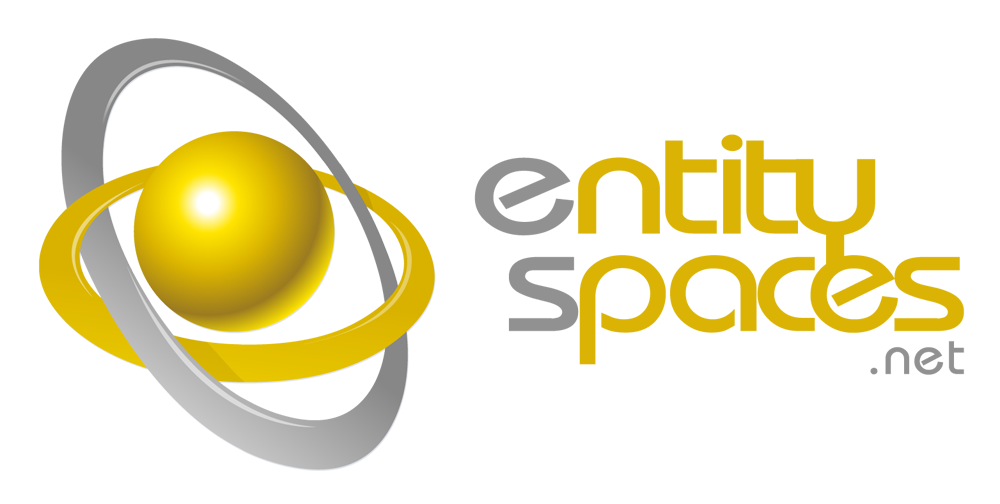
EntitySpaces Quick Reference

EntitySpaces Basic Usage

August 16, 2011  
EntitySpaces, LLC  
EntitySpaces Team

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

Persistence Layer and Business Objects for Microsoft .NET

Contents

[Introduction 1](#_Toc317167605)

[EntitySpaces Quick Start Checklist 1](#_Toc317167606)

[Configure 2](#_Toc317167607)

[Load 3](#_Toc317167608)

[Entity 3](#_Toc317167609)

[Collection 3](#_Toc317167610)

[Data Bind 4](#_Toc317167611)

[Save 5](#_Toc317167612)

[Hierarchical Model 6](#_Toc317167613)

[Dynamic Query 7](#_Toc317167614)

# Introduction

[EntitySpaces Studio](http://www.entityspaces.net/Portal/Default.aspx) includes the tools and assemblies needed to generate a persistence layer and business objects for Microsoft .NET, JavaScript, or both.

Feedback on our documentation is welcome. Please post comments, customer service, and support questions to our [Forums](http://www.entityspaces.net/portal/Support/Forums.aspx).

The **Getting Started** section of the documentation is designed for those new to EntitySpaces. A "**Getting Started**" PDF is, also, installed to your Start Menu. It introduces the EntitySpaces StandAlone, Visual Studio AddIn, templates, and walks you through each step for creating your first application with EntitySpaces.

The [walk-through video](http://developer.entityspaces.net/videos/ES2009_preview/ES2009_StarterApp.html) is recommended viewing for first-time users.

# EntitySpaces Quick Start Checklist

Each step of the checklist below is covered in detail in the "Getting Started" section of the documentation.

* Install EntitySpaces.
* Launch EntitySpaces
* Connect to your database.
* Generate your EntitySpaces classes into your Visual Studio Solution folder(s).
* Add the proper connection information to your app.config /web.config file.
* Add references to
  + EntitySpaces.Core
  + EntitySpaces.Interfaces
  + EntitySpaces.DynamicQuery
  + System.Runtime.Serialization
* Add a reference to **one** of the EntitySpaces Loaders
  + EntitySpaces.Loader (recommended for most applications)
  + EntitySpaces.LoaderMT (medium trust mode).
* Register the Loader.
* Add reference(s) to one or more of the EntitySpaces database provider(s). Just the ones you need.
* Add a using statement (Imports in VB.NET) for the namespace of your generated business objects (the default is BusinessObjects).

# Configure

Configure

# Load

You can populate EntitySpaces Business Objects (entities or collections) using the pre-generated methods on each, via EntitySpaces DynamicQuery API, or by creating methods in your custom classes.

## Entity

The Entity Represents only zero or one row in a Database Table. If you use Dynamic Query to Load an Entity, it must return at most one row; otherwise an exception will be thrown. If zero rows are returned with a call to entity.Query.Load, or entity.LoadByPrimaryKey, then the method will return false. If one row is returned, the Load methods will return true.

## Collection

The Collection is a collection of Entity objects. The Collection represents zero or more rows from a database table, each row is associated with an Entity object within the collection. When you insert/update/delete an Entity that is part of a Collection, then you must call Save on the collection, otherwise an Exception will be thrown. The Collection Load methods return false if zero rows are loaded, otherwise true.

# Data Bind

A few Windows Forms examples are included below. See the samples installed to your Start Menu for more.

# Save

Save

# Hierarchical Model

EntitySpaces uses the foreign keys in your database to automatically build the hierarchical object model. The Generated Master Template does this by default. Generating the hierarchical model may be disabled by unchecking the box on the Advanced tab.

EntitySpaces will automatically save down the hierarchical tree from Parent to Child. This process is iterative, so calling save on an EntitySpaces Northwind CustomersCollection (custs.Save()) will, also, save all changes to custs.OrdersCollectionByCustomerID, which will in turn save any changes to OrderDetailsCollectionByOrderID. If a Parent has an auto-incrementing key, then that is retrieved and filled in to the foreign key column for you. Everything is handled in the right order, so a Parent will not be deleted before the related Children, or added after related Child records. This prevents foreign key constraints from being violated in the database.

To prevent endless loops, EntitySpaces does not automatically save up the hierarchical tree. To help you determine whether a particular hierarchical sub-object is upstream or down, the default generated names for upstream sub-objects will start with the prefix "UpTo".

The hierarchical properties are lazy-loaded and cached. If, after changes are saved to the database, you need to refresh the cache, set the hierarchical property to null (Nothing in VB.NET). It will be re-loaded from the database on next use.

# Dynamic Query

The EntitySpaces DynamicQuery API may be used to load entities and collections. More advance queries may be constructed using the generated query objects.