Haystack Code Generator for .NET Version 7.0

The Haystack Code Generator for .NET (formerly known as the PDSA .NET Data Access Class Generator) allows developers to create classes to perform all CRUD (Create, Read, Update, Delete) actions against a database engine such as SQL Server or Oracle. Haystack is a template-driven code generator, meaning you can modify how code is generated by customizing pre-defined templates or by creating your own templates. Haystack has many pre-defined templates and the template language syntax is very straight-forward so you can modify them quickly and easily. Haystack is developed using the latest version and features of .NET to ensure that it is fast and easy to use.

Differences from DACGen Version 6.x

There are so many differences, that it is hard to even know where to start. This version is completely re-written using .NET 3.5/4.0. Below is a short list of the new features.

1. Provider-model driven
2. All templates stored in .TXT files instead of in an MDB file
    a. Allows for better source code control
    b. Easier customization
3. All new generation engine makes template customization much simpler
4. Completely new data model and validation model
    a. Can now separate how data is loaded from the validation
    b. The validation model can be used with any data methodology, it is not dependent on the data layer at all
5. Support for generating XML data classes.
6. Support for generating WPF, Silverlight and ASP.NET add, edit, delete forms.
7. Service Oriented Architecture (SOA) model
8. New flag supports dynamic SQL and stored procedure generation usage in the same class

Of course, there is much more in this new version as well.

Robust Data and Validation Model

Our data model and validation system, by default, generate the following classes:

1. Data transfer object (useful for WCF applications)
2. Validation class (automatic generation of validation rules picked up from your database table)
3. Business logic class (a place for you to add your own custom validation rules)
4. Data access class (CRUD (Create, Read, Update, Delete) logic)

info@pdsa.com
Toll Free: 888.899.7372
714.734.9792

Our data access classes use a provider based data layer so switching from one database engine to another is quick and easy. No code required!

Haystack provides a unique, flexible approach to data access, validation and business logic. Using the templates provided with Haystack, the tool creates an "entity" class (or data transfer class) that just represents data (no methods), a set of data access classes that are used to populate that entity class, and a set of validation classes used to validate the data in the Entity class prior to submitting the data to the database. Of course, Haystack supports full generation of both C# and Visual Basic.

SOA Model

Haystack can generate classes that support an SOA model using Windows Communication Foundation (WCF).

WPF Window/User Control Generation

We have templates to support WPF Window and User Control Generation using a Model-View/View-Model architecture. We give you a complete set of data classes hooked up to a complete set of WPF user controls for an add/edit/delete window in WPF.

Silverlight User Control Generation

We have templates to support Silverlight User Control Generation using a Model-View/View-Model architecture. We give you a complete set of data classes hooked up to a WCF service that allows you to interact with the generated Silverlight User Controls.

XML File Data and Validation Model

If you do a lot of XML file processing you will appreciate Haystack's XML data access class generation. Just like data classes for database tables you have the ability to read element-based XML files and create validation and data access classes to read and write XML files. Now, creating a CRUD application for your XML files is just as easy as for your database tables.

Dynamic SQL / Stored Procedures

All data class generation support Dynamic SQL and/or Stored Procedures. You may choose one or the other, and you can even choose both. Whether using dynamic SQL or stored procedures, command parameters are used for the passing of parameters. This ensures consistency between the two methods and eliminates potential SQL injection attacks.

Settings Storage

All generation templates are stored in text files. All Haystack default settings are stored in XML files. All your schema data and property settings are stored in a SQL Server 2005/2008 or SQL

info@pdsa.com
Toll Free: 888.899.7372
714.734.9792

Server Express 2005/2008 database. Access to your property settings is managed using a provider model that makes switching project storage very easy. By using text and xml files you get many benefits:

1. Versioning and management of templates using source control.
2. Ensures consistency among developers by centralizing templates and settings.
3. Saving your project settings in a database engine gives you speed and a central backup option.

Licensing

Haystack Version 7.0 is licensed for a single-user on a single machine. This model allows users to purchase additional copies of Haystack and register their copy all via the internet.

Supported Platforms

Haystack is designed to work with the following technology stack.

- Visual Studio 2008/Visual Studio 2010
- Microsoft .NET 3.5 and Microsoft .NET 4.0
- SQL Server 2005 / 2008 for storage of Haystack meta-data
- SQL Server 2000 and above can be used to generate code from.

Of course, the templates can be customized to support previous versions of .NET and Visual Studio, as well.

Future Enhancements

PDSA will provide support for Oracle 10x shortly after shipping version 7.0.

Migration of Old DACGen Projects

At this time, we are not sure of the migration path of projects from prior versions of our DACGen. Of course, there is nothing wrong with keeping your old data classes as they work quite well. In addition, the old classes use a different DLL and namespace from your newly generated classes, so you can use the old classes and new classes within the same project.

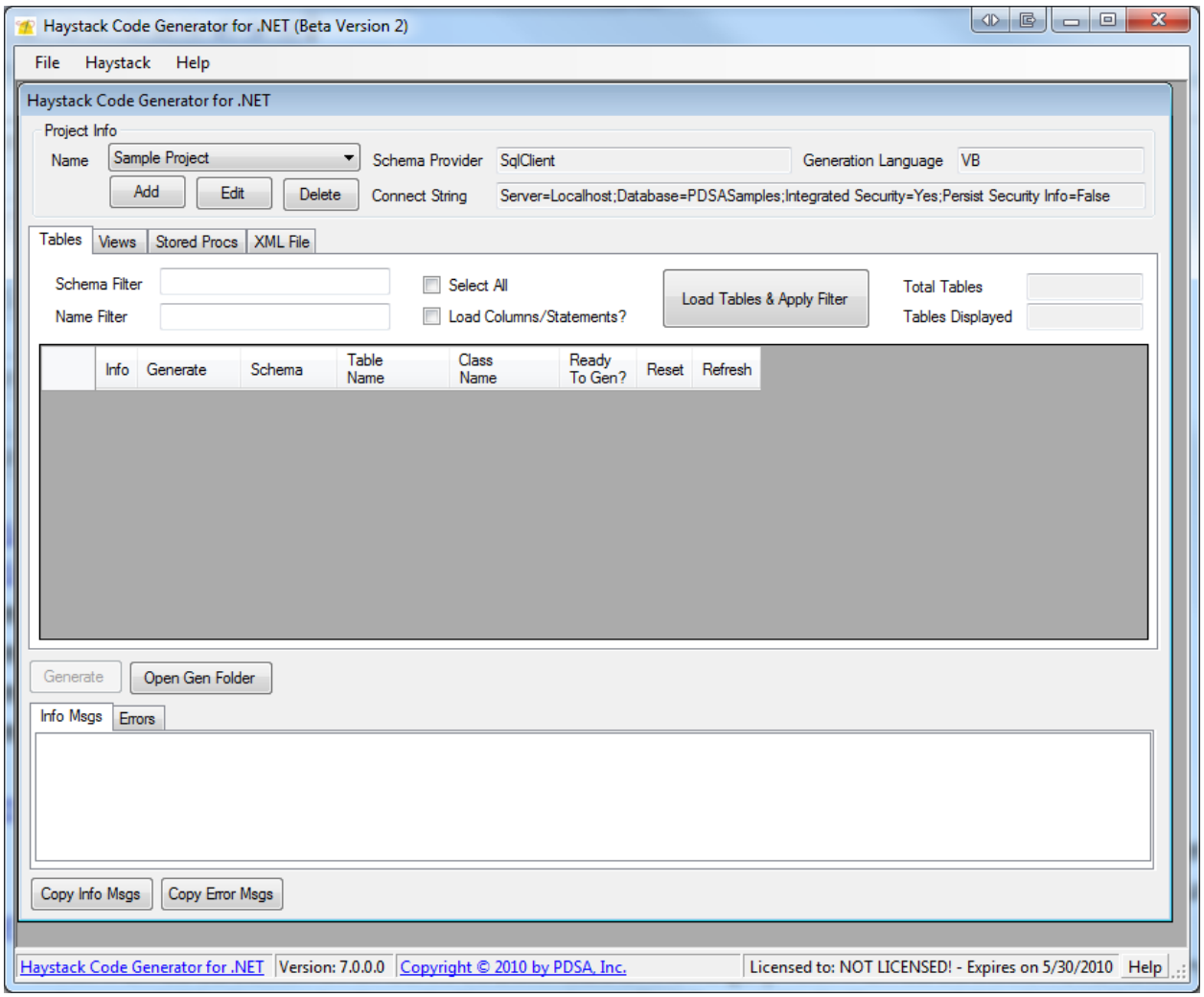
Sample Screens

Below are some examples of what the new version will look like. Of course, at this point, these are subject to change, but this should be fairly close to what it will look like.

info@pdsa.com
Toll Free: 888.899.7372
714.734.9792

Schema Display Screens

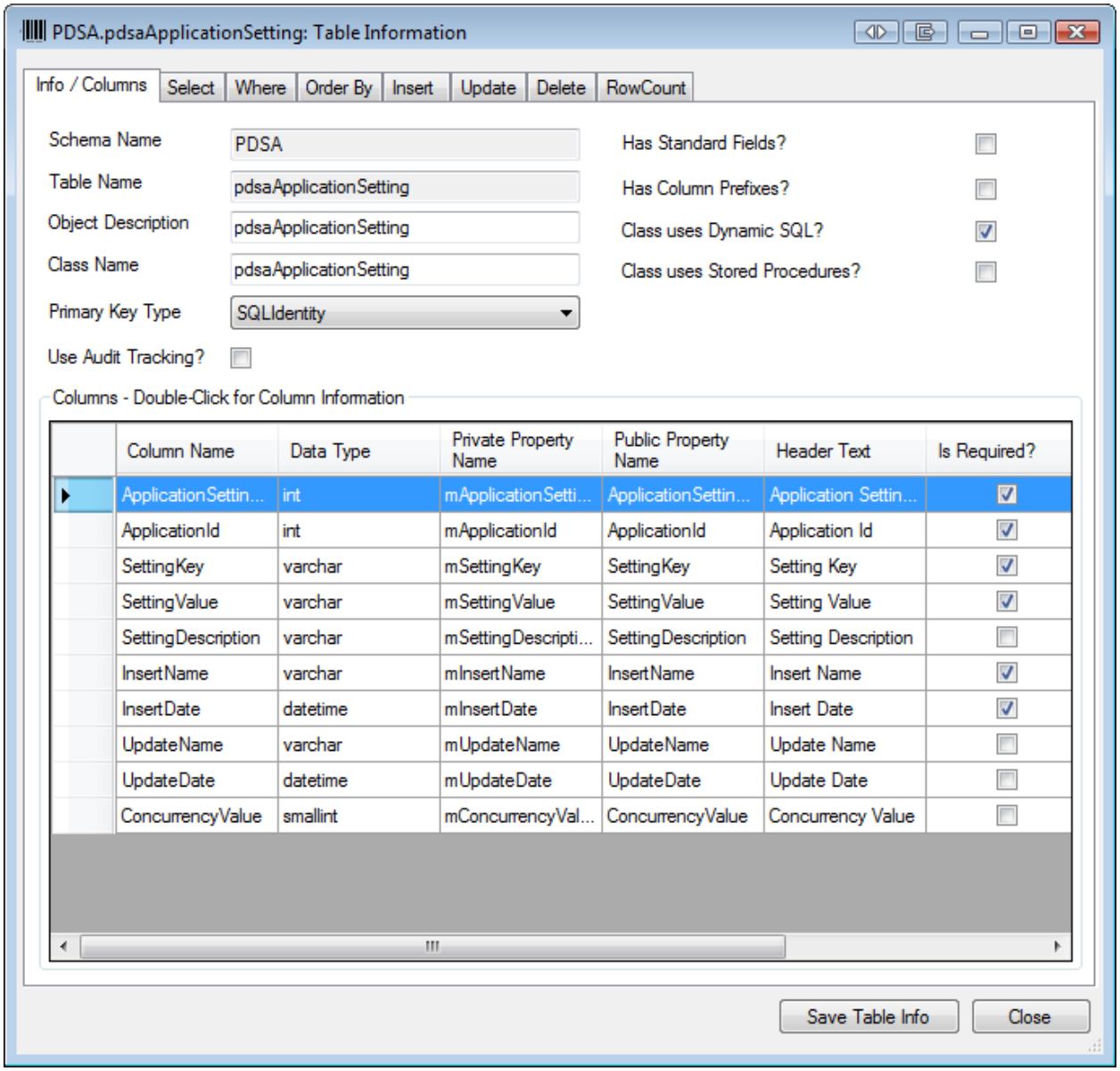
You now have the ability to filter the tables, views and stored procedures you read in by schema or the name of the object.



Sample CRUD Generation Screens

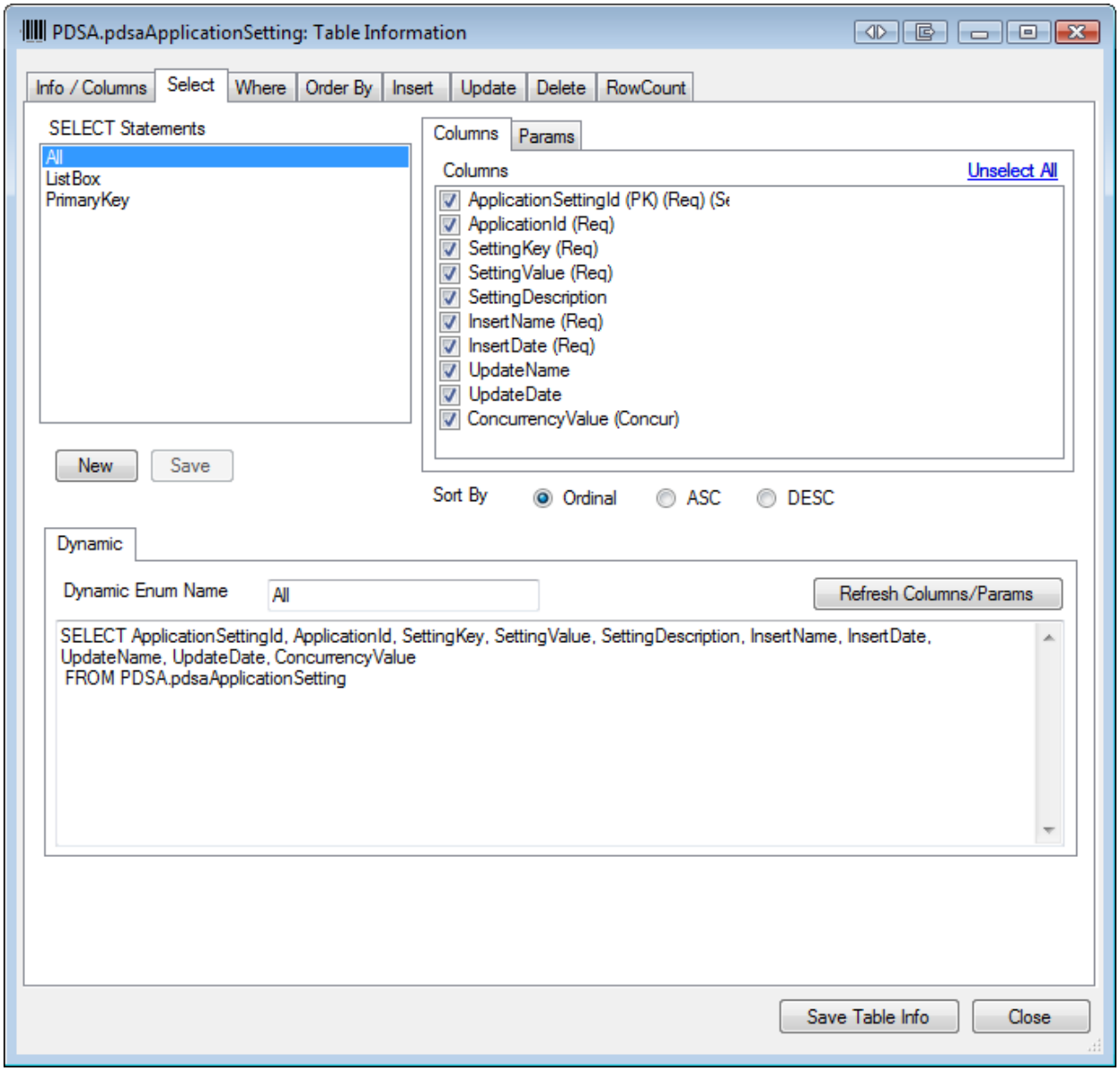
After selecting an object, all columns, selects, where clauses, order by, insert, update, delete and row count statements are pre-generated. All these statements are generated from templates, so modifying them can be done prior to, or after this screen is displayed.

info@pdsa.com
Toll Free: 888.899.7372
714.734.9792



Below is an example of how the SELECT statements look in Haystack.

info@pdsa.com
Toll Free: 888.899.7372
714.734.9792



info@pdsa.com
Toll Free: 888.899.7372
714.734.9792

For More Information

Got a problem? Call us for a free analysis of your requirements to see what products or services may be the best fit for you. Our business acumen coupled with our technical and practical experience enables us to help you meet your business goals on time and within budget.

PDSA provides solutions for the real world. We offer a full range of products and services:

- Custom Software Development
- Training and mentoring
- Application, security and database audits
- Specialized partnering
- Business process engineering
- Proven software development methodology
- Technical excellence in Visual Studio, SQL Server, Web/Desktop/Services
- A full range of productivity products

Contact us:

- Call 1-888-899-7372
- Visit www.pdsa.com
- Email info@pdsa.com