# logo-sql08.gif

# Microsoft® SQL Server® 2008 Express: A Free, Powerful Database for Web Development

**Writer:** Graeme Malcolm and Alistair Matthew (Content Master)

**Technical Reviewers:** Eric King, Zachary Owens, Mike Wachal

*Note: A series of short “How To” videos complements this article, and can be found* [*here*](http://www.msdev.com/Directory/SeriesDescription.aspx?CourseId=105)*.*

## Introduction

When you create a Web site, you frequently need a place to store data. It might be data input by visitors, information about products or services, news articles, discussions, or a whole host of other possibilities. Therefore most Web sites read and write data to and from a database and that doesn’t seem likely to change any time soon.

There are a lot of database systems available: some cost thousands and others are free and there is a similarly broad range of features on offer. Microsoft is well known for its advanced database solution – Microsoft® SQL Server® 2008. It’s a fully featured database server that scales to the largest sizes, supports highly available and fault-tolerant configurations, and includes top-specification analysis tools. That’s great if you’re building a business critical application for thousands of simultaneous users, but may be a little over the top for developing a Web site for a small hardware store – right?

Actually, SQL Server 2008 is available in a number of editions to suit all kinds of scenarios. For developing Web applications, SQL Server 2008 Express Edition is an ideal choice. It provides exactly the same advanced database engine as the other editions of SQL Server, with a few scalability restrictions (it can only make use of a single processor and 1 GB of RAM, and databases are limited to 4GB in size). The database engine is completely compatible with other editions of SQL Server, so even if your customer or hoster requires the scalability of SQL Server Web, Standard, or Enterprise Edition, any database application you build on SQL Server Express Edition will work unchanged when deployed to the production environment. (Note, in particular, that SQL Server 2008 Express matches the features of SQL Server 2008 Web Edition. This is the edition likely to be in use at your hosting provider, so you are unlikely to find you’ve used a feature that your hoster doesn’t support.)

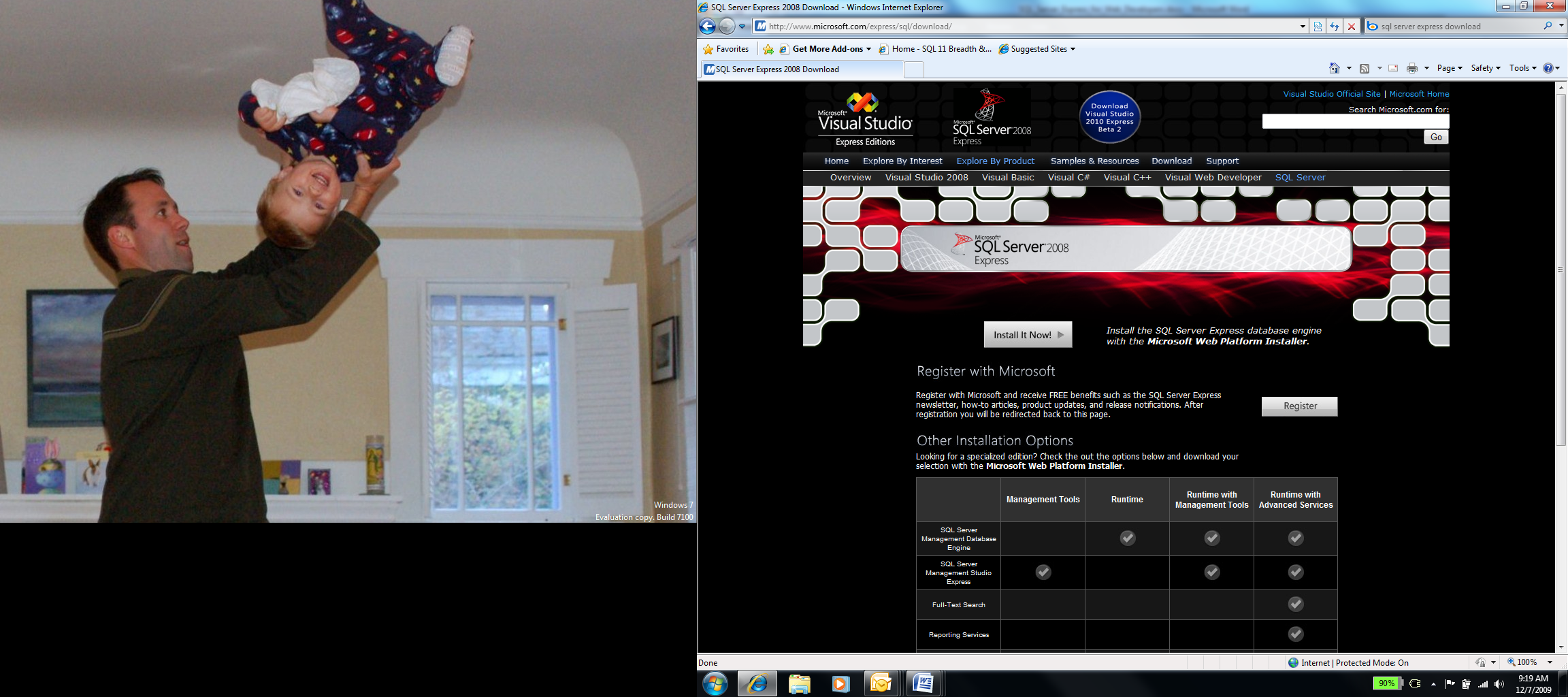
SQL Server Express Edition is closely integrated with Microsoft developer tools such as Visual Studio and Visual Web Developer Express, and it even supports advanced data types such as native XML, spatial data, FILESTREAM data and so on. In fact it’s surprising how many features of SQL Server are included in SQL Server Express when you consider its most important feature; it’s free – for both development and deployment!

There are three versions of SQL Server Express Edition you can choose from:

* SQL Server 2008 Express Edition with Tools: In this version you get the database engine with the SQL Server Management Studio Basic – this tool is all you need to create, edit, and manage databases.
* SQL Server 2008 Express Edition with Advanced Services: In this version you get the engine, the Management Studio, plus Full-text Search for searching text-intensive data, and Reporting Services for creating powerful reports on your data.
* SQL Server 2008 Express Edition (Runtime Only): In this version you only get the database engine.

You can also get the SQL Server Management Studio 2008 Express as a separate download. This might be useful to add where you have the Runtime Only edition, or to install on a separate computer for remote administration.

You can download any of these for free from <http://www.microsoft.com/express/sql/download>

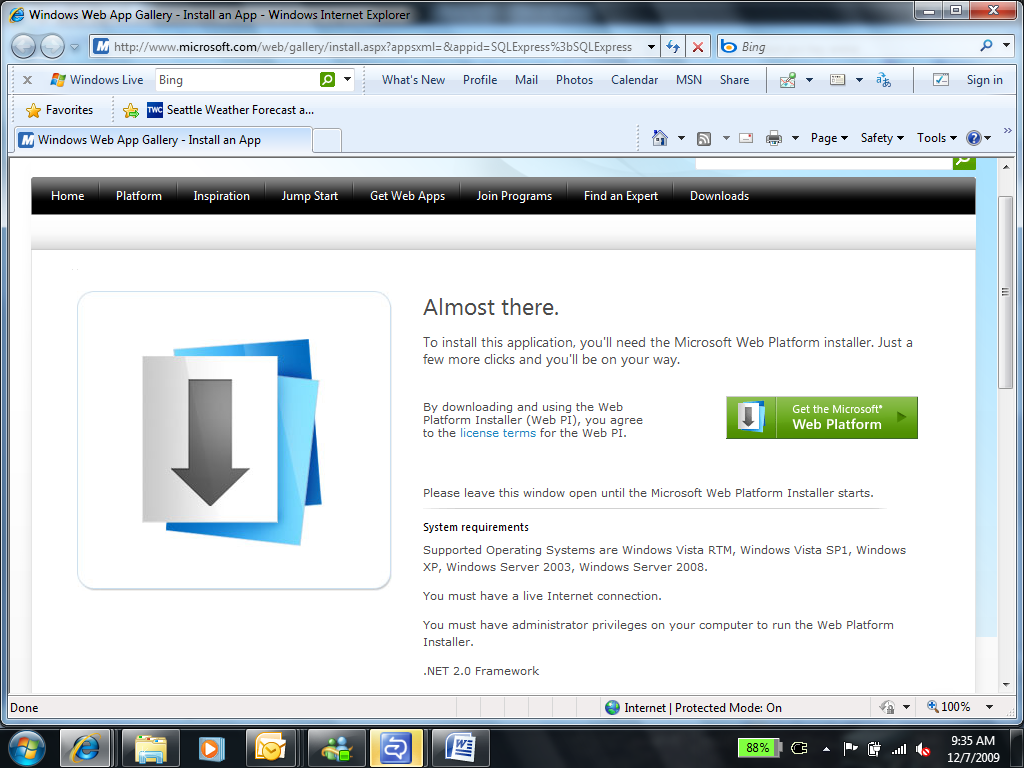


## Installing SQL Server Express Edition

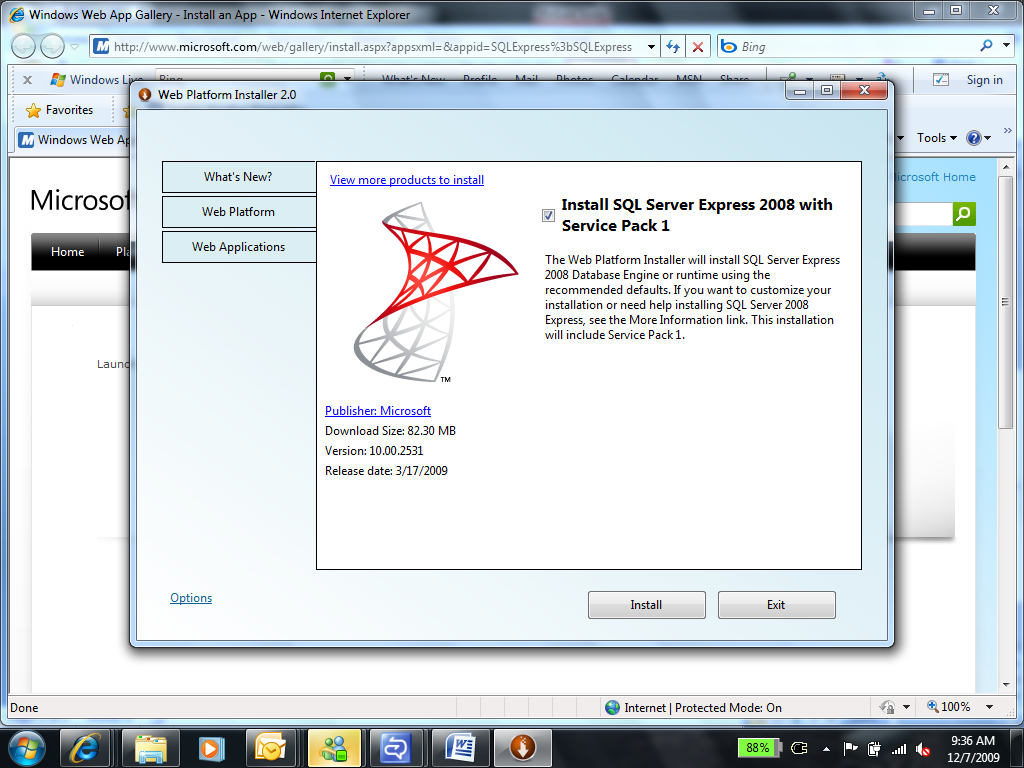
If you are new to SQL Server, the easiest method of installation is via the Web Platform Installer – also known as the Web PI. This tool, which launches automatically from the Website above, streamlines the installation of SQL Server. For instance, if you need to install any pre-requisites, Web PI takes care of this for you.

Web PI also can be used to install Microsoft’s entire Web platform, if you wish, including SQL Server Express, the IIS Web server, the .NET Framework, and Visual Web Developer 2008 Express. When you use this tool there is just one simple wizard to complete for all the products in the platform. Additionally, the tool downloads the latest version of each product, and the latest critical patches, and installs them for you, so you can be sure your platform is right up to date. In this example, we’ll just stick with SQL Server Express.

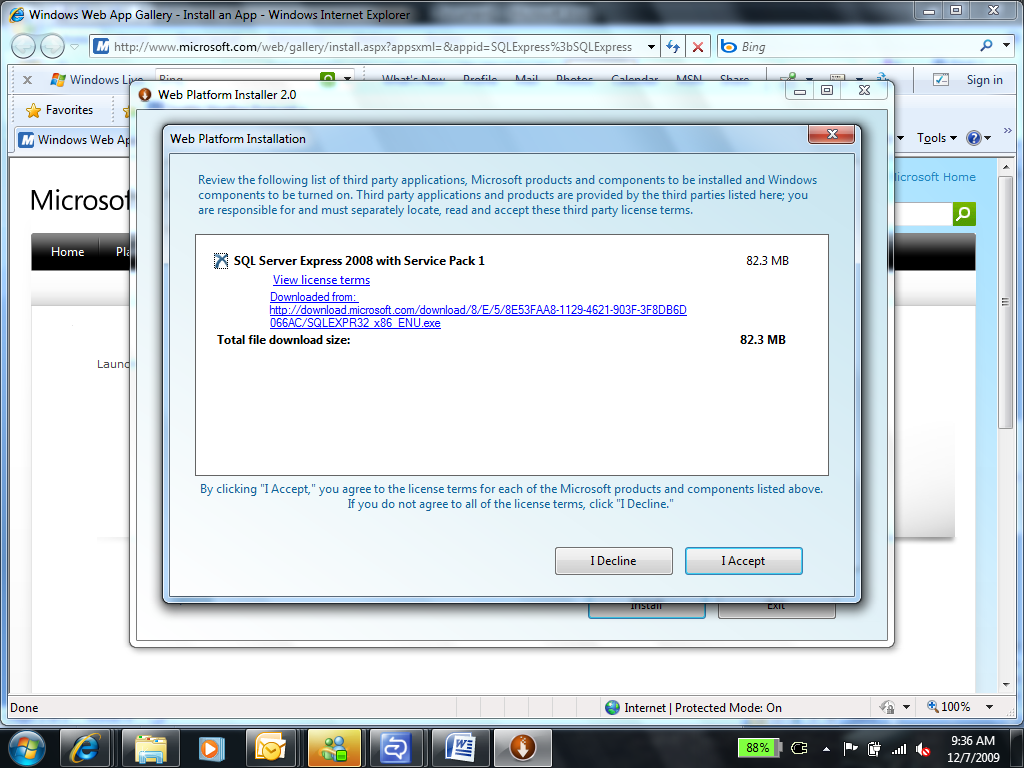
When you click on “Install” from your selected Edition at the Web page above, you’ll be asked to install the Web PI – go ahead and install this tool.



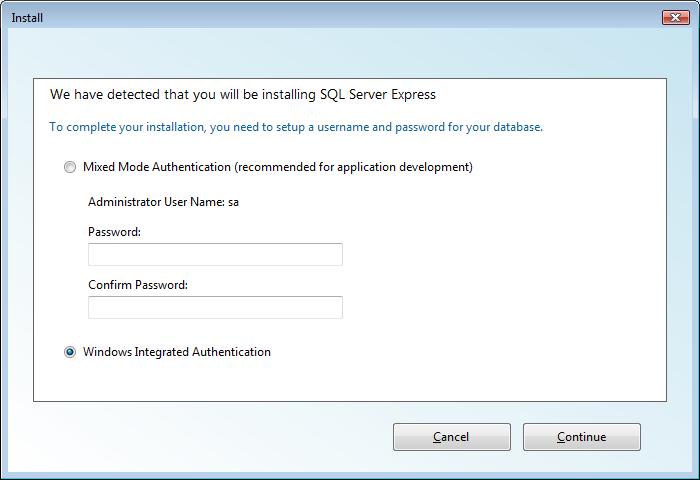
Upon installation, you’ll be brought to the SQL Server Express installation page.



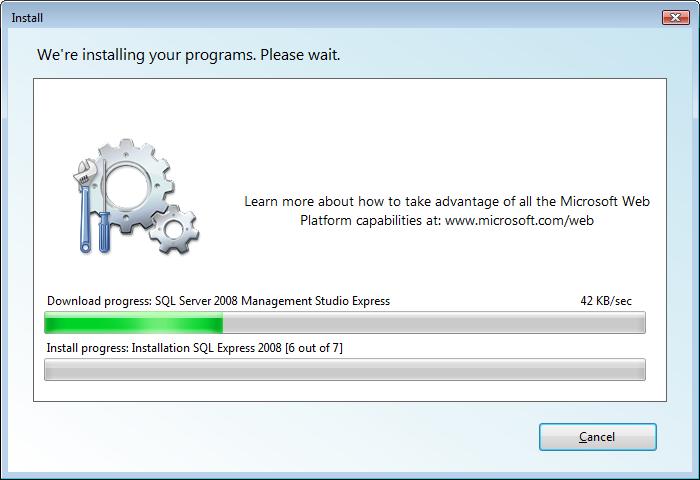
Click Install. You have to accept the relevant licensing terms:



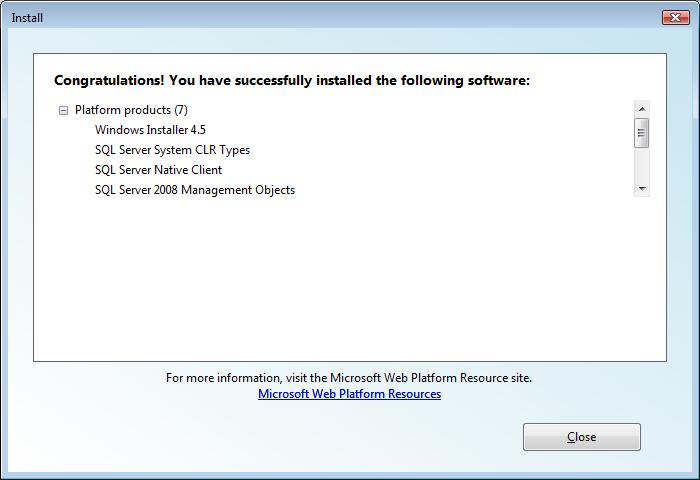
You also have to specify the authentication method to use:



Then the Web PI downloads and installs all the prerequisites and components for you:



This may involve a reboot, in which case the Web PI continues after you’ve logged on. When the process is complete you see a summary page:



## Creating a Database

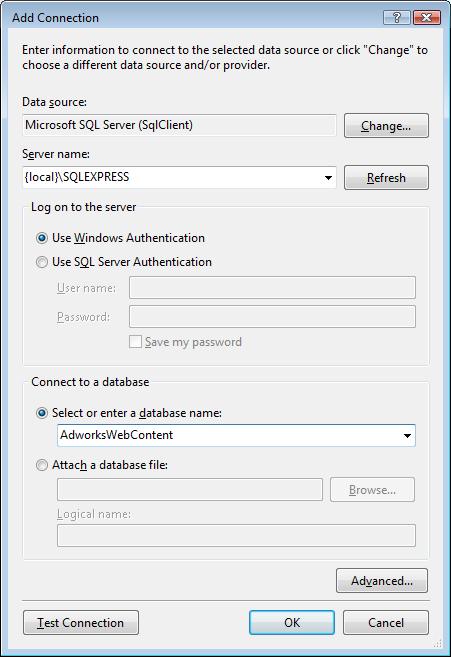
If you’ve installed a version of SQL Server Express Edition that includes SQL Server Management Studio, you can use it to create your databases. However, in all three editions you can create and configure a database from within your preferred development tool: Visual Studio or Visual Web Developer Express. This is one of the advantages of using a development tool that is so tightly integrated with the database server.

In fact there are two ways to create a database for a Web site with different architectures.

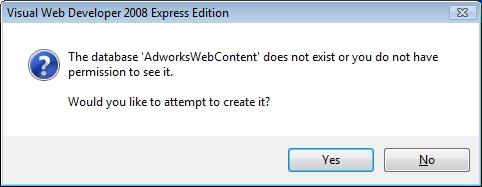
* Create a database in the local instance of SQL Server Express.

This approach most closely matches the way the database runs at most hosting providers, but it requires that you are running the development environment as a user who has permission to create database in the local instance of SQL Server Express. If you develop software using an account with minimal permissions, you can use SQL Server Management Studio or the SQLCMD command-line tool to create a SQL Server login and grant the necessary permissions.

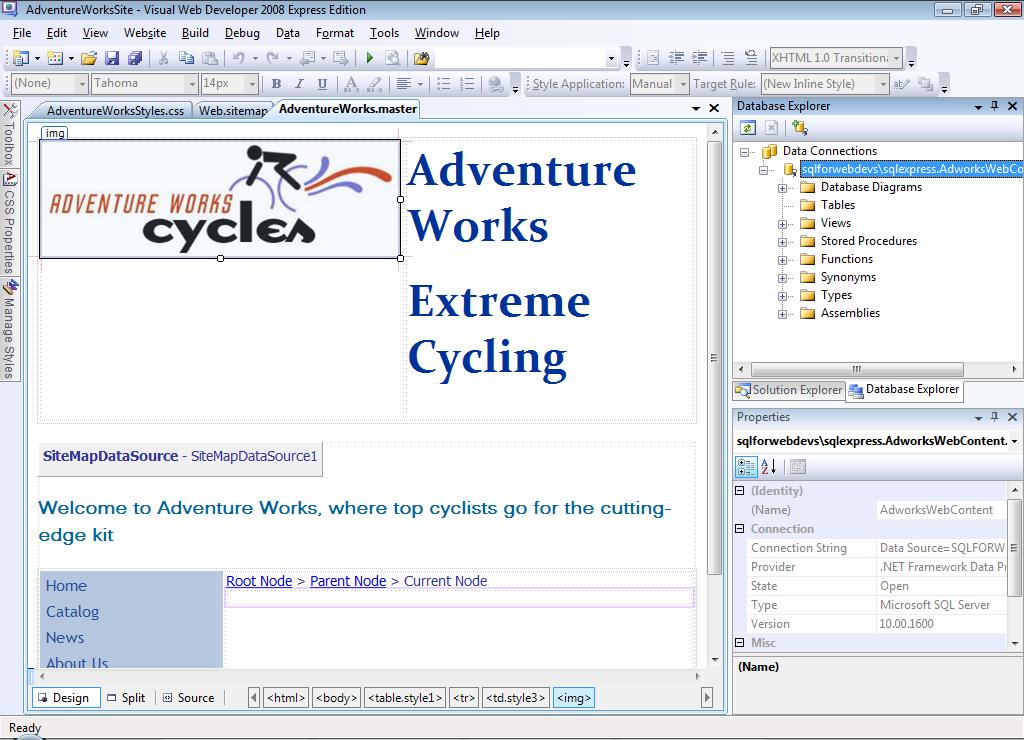
On the tools menu, click **Connect to a Database** and then choose the **Microsoft SQL Server** DataSource. The Add Connection box appears:



In the **Server Name** box you need to enter *ServerName*\*InstanceName* and you can use {local} for databases on the development computer. To create a new database enter a new name under **Select or enter a database name**. When you click **OK** you’ll see this dialog:



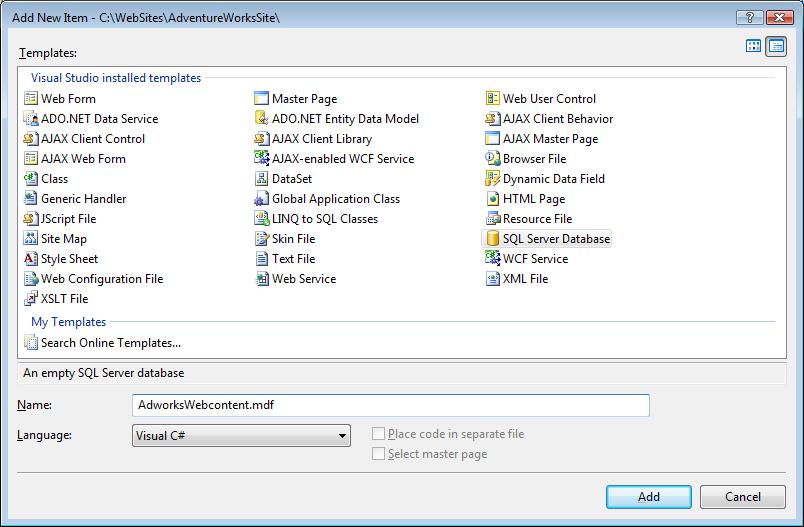
The new database is created and the Database Explorer pane appears in the development environment:



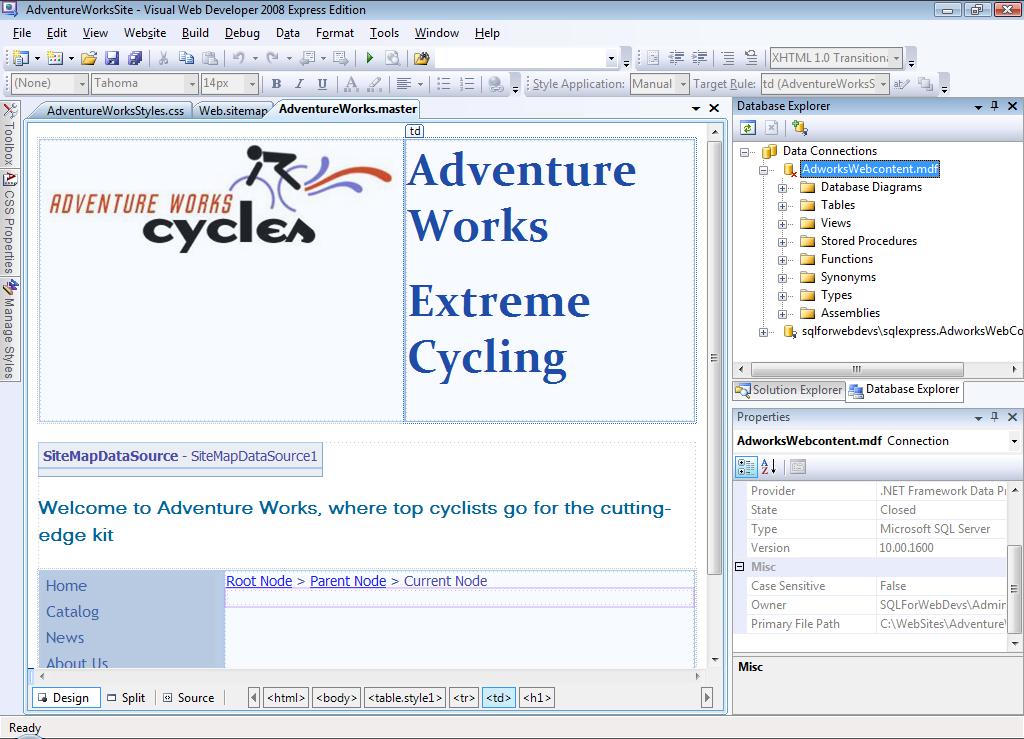
* Create a database that is saved as a file in the application folder itself.

With this approach, the database is created as a file and attached to a user-instance of SQL Server Express dynamically at run time. This removes the requirement to have administrative rights in the local SQL Server Express instance and makes it possible to distribute the database along with the rest of the Web application project files, so you can work on the project on any computer with SQL Server Express installed. However, you should be cautious when using this approach as it does not reflect the way that database are typically configured on hosted Web sites, so you might need to perform additional testing before deploying the application to production.

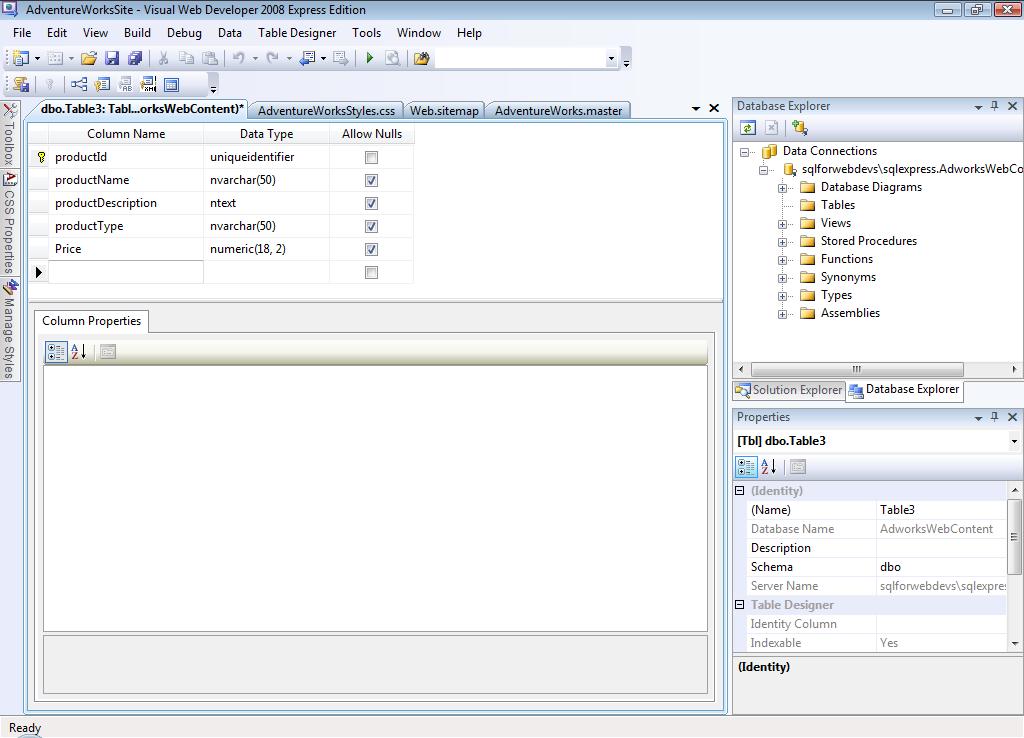
To use this approach, add a new item to the site, and choose the **SQL Server Database** template:



When the process is complete you can see the new database in Database Explorer like this:



After you’ve created your database, you can start adding tables, views, stored procedures and other features, all in your development environment. For example, to add a new table, right-click **Tables** and then click **Add New Table**. You can add columns, set data types, and set primary keys in a simple GUI:

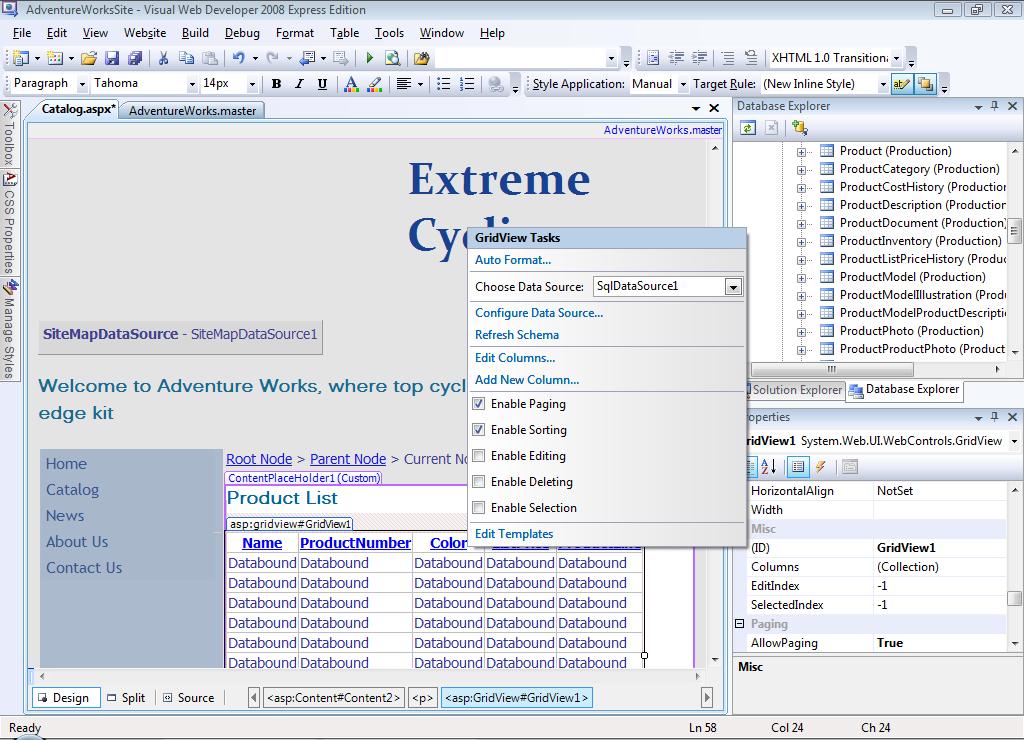


## Creating a Data-Bound User Interface

Of course you must present the data stored in SQL Server Express on your Web pages and frequently you must write data from those pages to the database. There are several approaches to accomplishing these tasks.

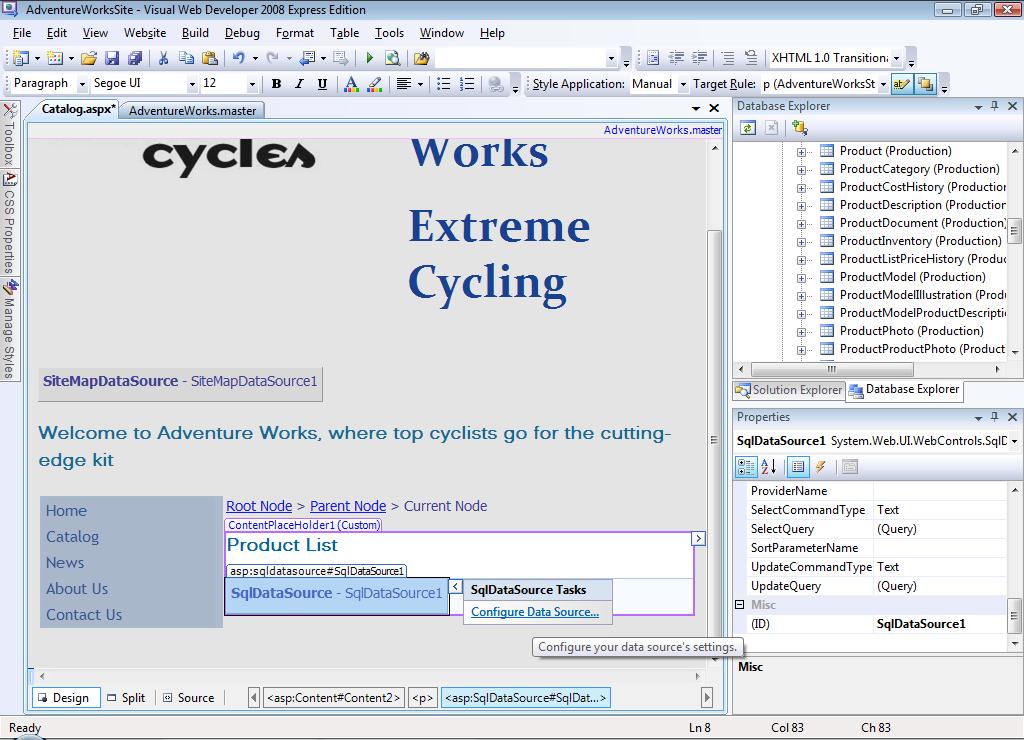
First, you can write your own code to read and write data; this gives you a large degree of control over the process but it takes a lot of work and debugging time. Of course, you could place this code in a .NET library, so you could reuse it in other Web sites. You’ll probably use ADO.NET objects in this code to interact with the database and its stored procedures.

If time is short, however, it’s really quick to create data-bound control on Web pages in Microsoft development environments. For example, you can create a grid control on the Web page simply by dragging a table or view onto it. When you do this, the Grid View Tasks menu appears, which you can use to configure the columns to display and other feature of the table:

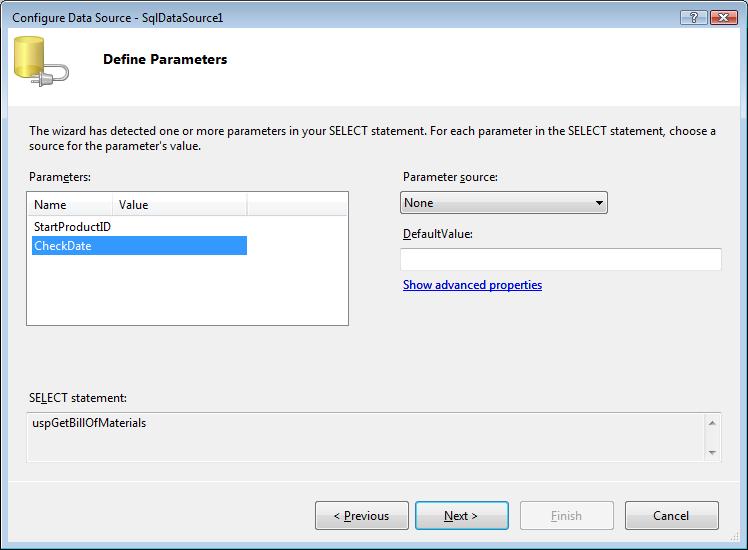


If that’s too simple, you can gain extra flexibility by creating a data source from a stored procedure or SQL statement. Then you can bind multiple controls to it. This is a good way to display similar data on several pages.

You can drag a SQLDataSource control from the Data section of the toolbox onto your Web page. Then, in the SQLDataSource Tasks list that appears, click **Configure Data Source**.

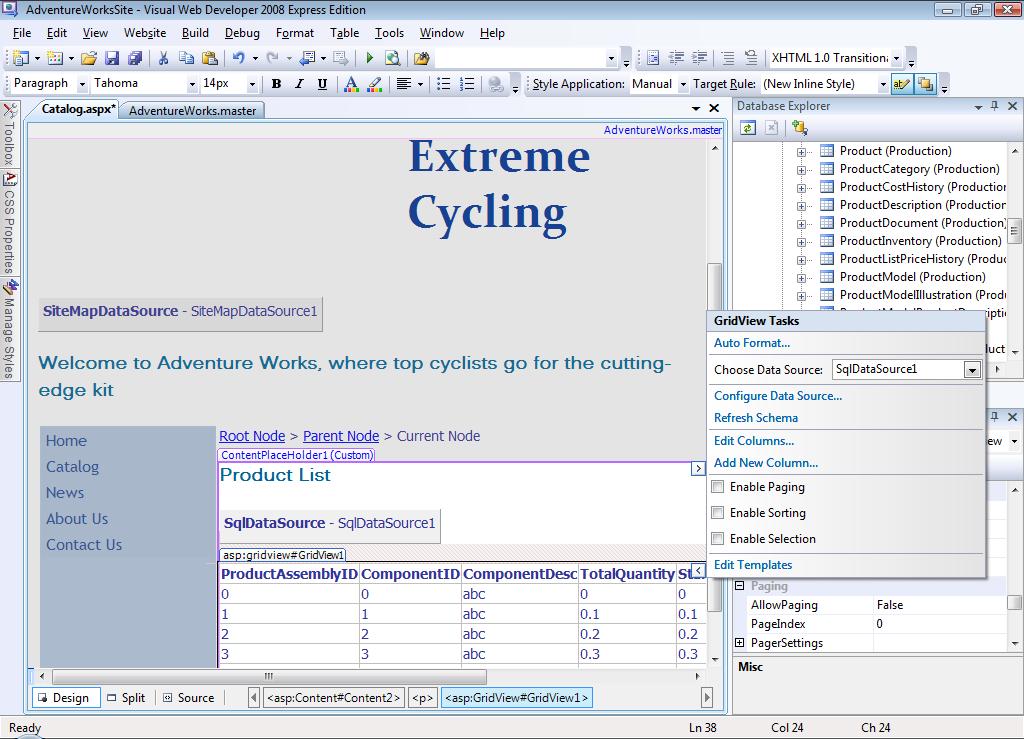


This starts a wizard in which you specify the table, view, or stored procedure you want to use, and other information. For example, if you specify a stored procedure, you may be asked for parameters:



The wizard also lets you test the query to make sure you get the results you expect.

After you’ve created a data source control you can bind many different types of control to it. For example, here’s a GridView control being bound to a data source called “SqlDataSource1”:



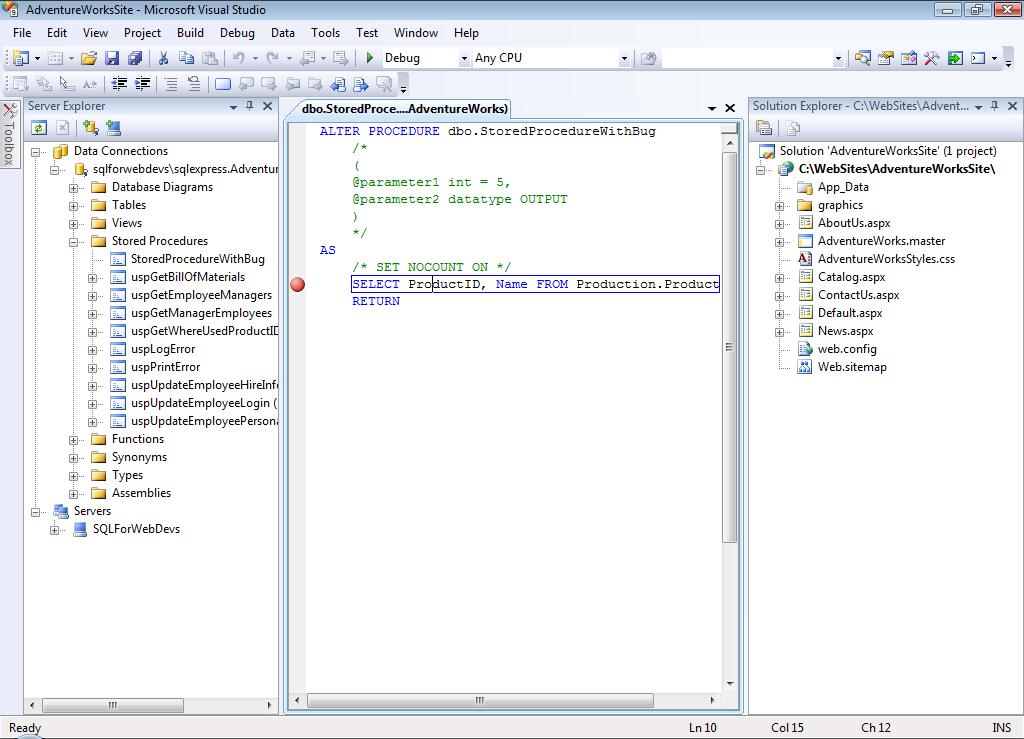
Several controls in the .NET Framework allow you to edit data, and you can create paged and sorted displays rapidly.

## Debugging Data Access Code

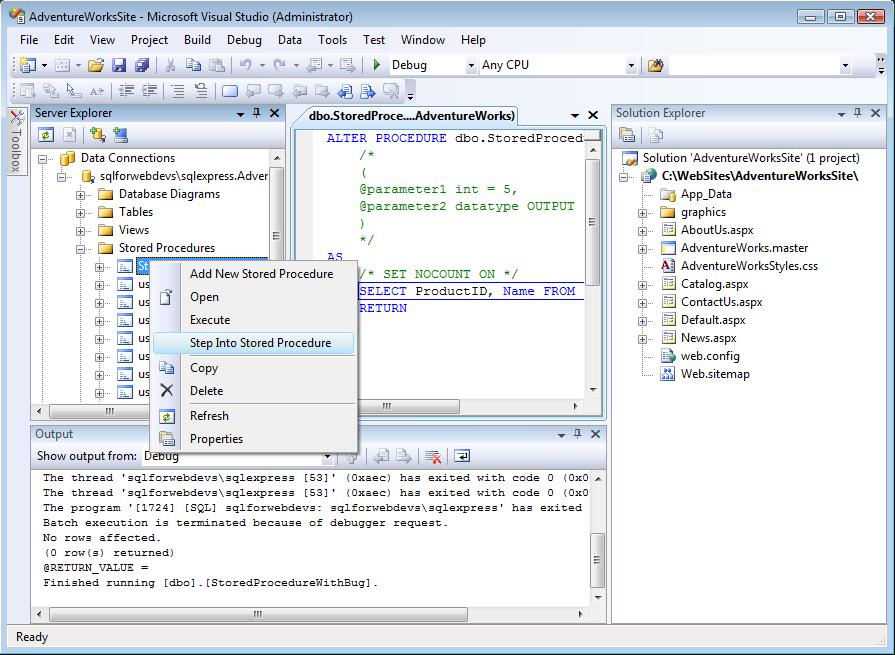
Microsoft Visual Studio and Visual Web Developer both include powerful debugging tools that step through code line by line and enable you to examine variables, errors, and other runtime aspects. However, with most databases, when the script in a stored procedure is executed, the debugger loses control – you only get full debugging functionality when the procedure is complete and control returns to the .NET Framework code.

This is not true when you use SQL Server 2008 Express: In Visual Studio Professional Edition the debugger can step through Transact SQL code for a stored procedure just as if it was managed code in the Web site itself. This enables you to spot bugs in stored procedures that would otherwise take much longer to diagnose.

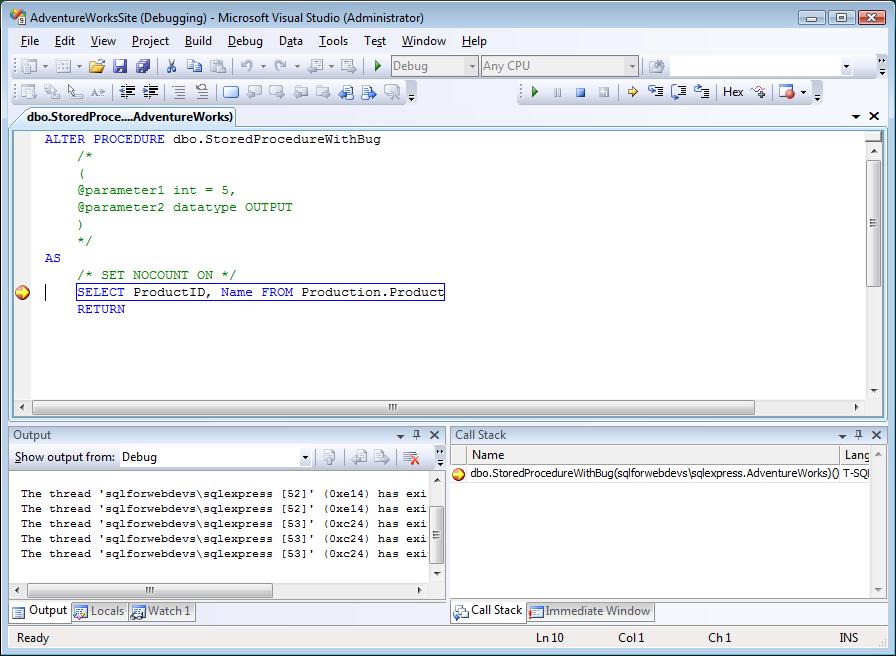
When you open a stored procedure you can set break points in it, just as you would for C# or Visual Basic code:



Now, start debugging the stored procedure by right-clicking it and click **Step Into Stored Procedure**:



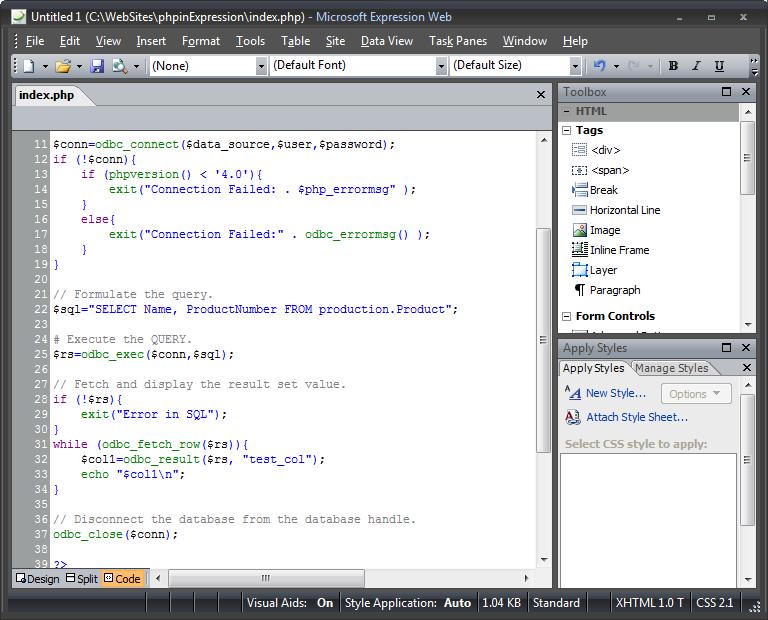
The debugger starts and you get full access to the call stack and other tools:



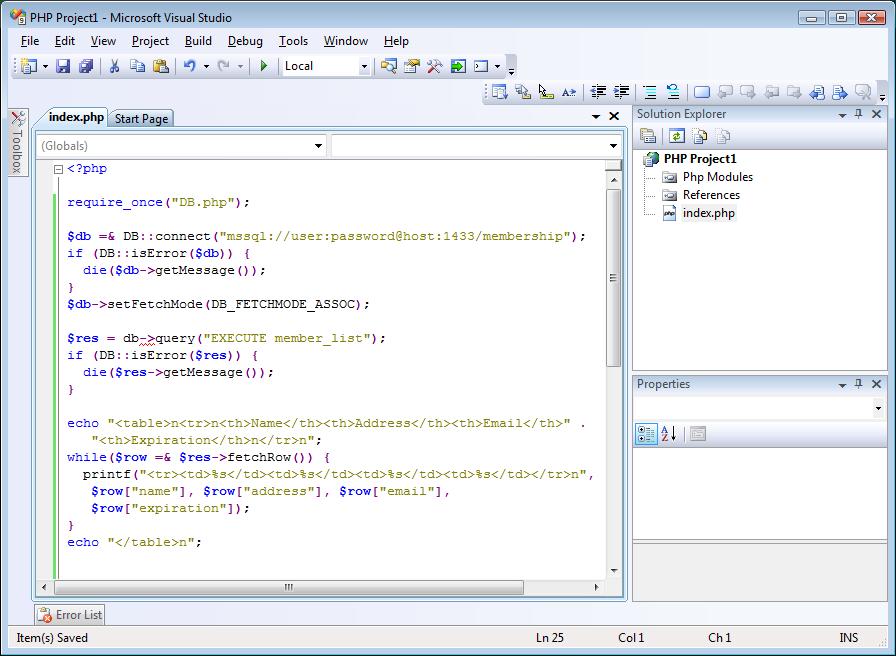
## Accessing Data with PHP

So far in this article, we’ve discussed development in Visual Studio or Visual Web Developer with .NET Framework languages such as Visual Basic and Visual C#. However it’s also true that you can use other server-side technologies to access data stored in SQL Server Express and other development environments. For example, PHP developers can access SQL Server databases by using the SQL Server Driver for PHP (available here: <http://www.codeplex.com/SQLSRVPHP>). Furthermore, you can use whatever development environment you prefer, from Notepad up, to write code for these technologies.

For example, in the following screenshots a developer is writing PHP code in the Microsoft Expression Web development tool to access an SQL Server Express database. You could use a third-party development tool in exactly the same way.



You can even write PHP code within Visual Studio – JCX Software provides the VS.Php add-in with advanced functionality such as debugging and IntelliSense:



I haven’t space in this article to dive deeply into PHP and SQL Server Express, but you should know that, once you’ve set up the driver, writing PHP against SQL Server is very similar to writing PHP against any other database and the objects are very well documented.

The driver is available at this link: <http://www.codeplex.com/SQLSRVPHP>. Further, you can find a training kit on PHP and SQL Server here:

<http://go.microsoft.com/?linkid=9656081>

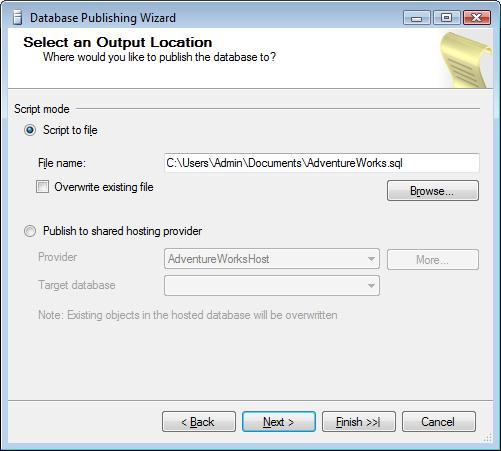
## Deploying a Web Site

When you have completed, tested, and debugged your Web site you must then deploy it to the production server – usually one run by your hosting provider – and you must deploy the database. In practice the database deployment method depends on the provider you use.

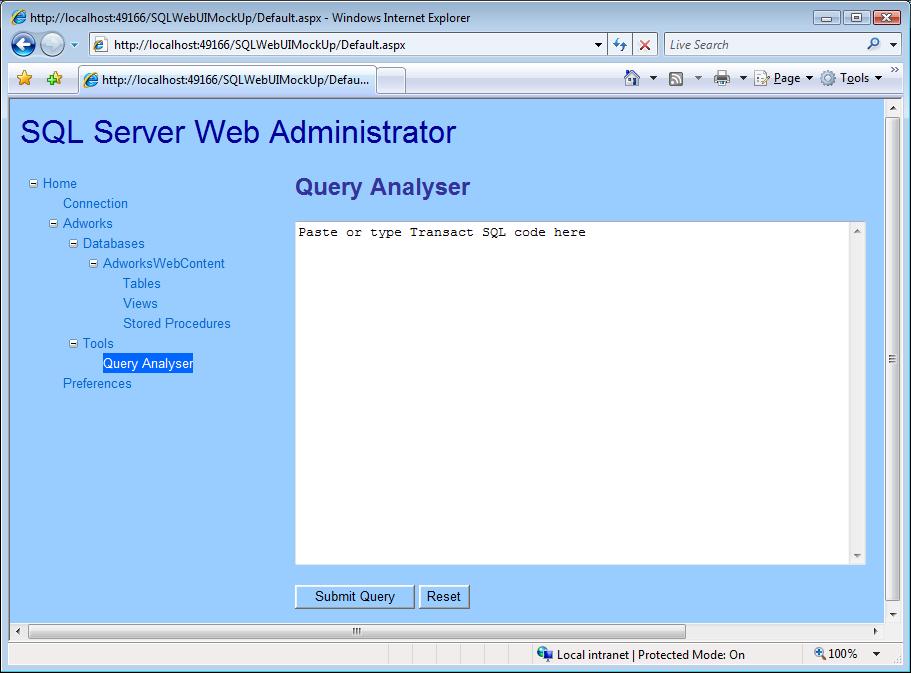
Common methods for database deployment include:

* Use the Data Publishing Wizard to generate a Transact SQL script

The wizard can generate a script that creates and populates the database when you run it on the hosted database server.



To run the script some hosting providers have a Web-based Transact SQL script editor. You copy and paste your script into this page and execute it.



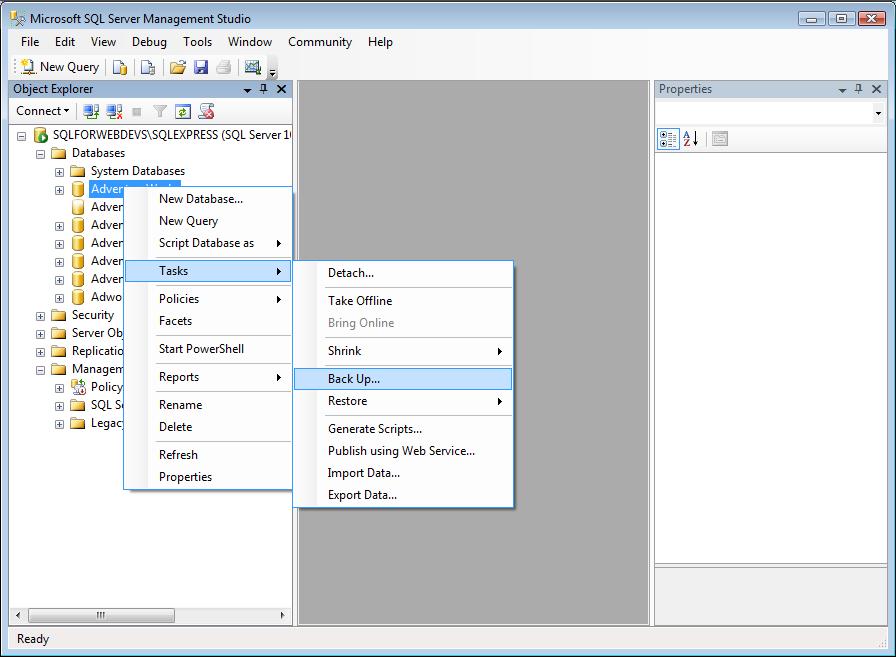
Some providers give you other ways to run such a script. For example, they may enable you to connect SQL Server Management Studio Basic to the hosted database server.

* Upload the database as a .mdf file

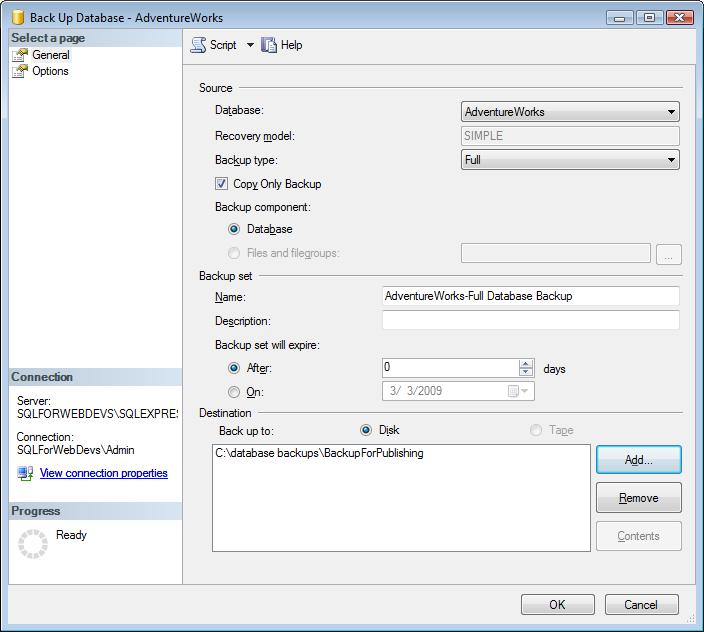
In this method you copy the .mdf file to a location on the hosted server. Then you must attach it to the SQL Server instance. You might do this by using a custom Web user interface.

* Backup and restore the database

In this method you use SQL Server Management Studio to backup the database:



Make sure you do a Full backup and select **Copy Only Backup** to avoid disrupting your normal backup routine:

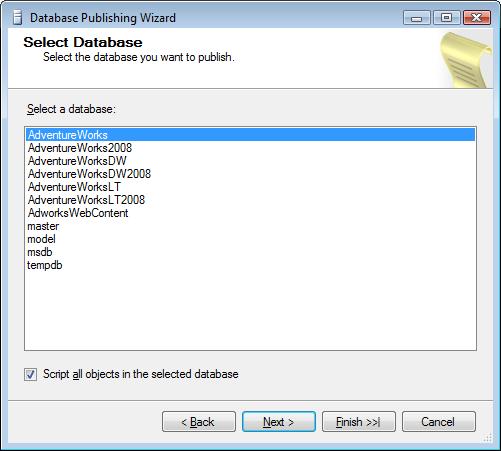


The backup should proceed smoothly and show a confirmation message.

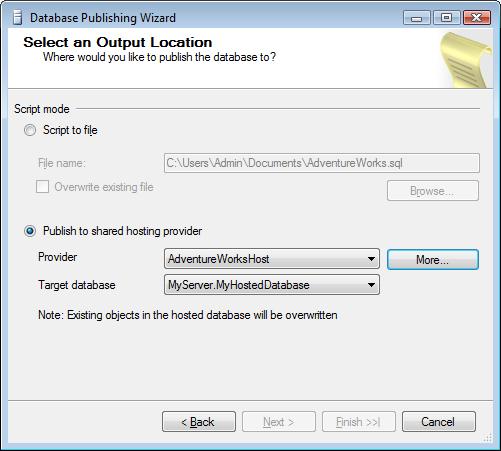
Next, upload the .bak file to the hosted server. You must then restore it to the hosted SQL Service instance. The provider may have a custom Web user interface for this purpose.

* Use the Database Publishing Wizard with the Data Publishing Service.

This wizard is including with Visual Studio and Visual Web Developer. Your provider must run the Data Publishing Service on the server for this method to work. Start the Data Publishing Wizard by opening the Database Explorer pane, then right-click the database and choose **Publish to Provider**. First you must choose the database to publish:



Then enter the provider details and the name of the remote database. Your hosting provider will provide these details, including credentials with which to authenticate against the service:



Whichever method you have used, you should now have the database on the production Web server at the hosting provider. However, the connection string that you used to connect the Web site to the database during development will not work at the hoster: you’ll need to edit that string before the Web site can run. To edit the connection string open web.config on the production server and find the <connectionStrings> tag. Here’s an example:

<connectionStrings>

<add name="AdventureWorksConnectionString1"

connectionString="Data Source=*servername*\*instancename*;Initial Catalog=*databaseName*;Integrated Security=True" providerName="System.Data.SqlClient" />

</connectionStrings>

You must ensure that the *servername*, *instancename*, and *databaseName* strings match the details supplied by the hosting provider.

# Summary

You’ve seen how capable SQL Server 2008 Express can be and how it can help you write Web sites faster. In particular:

* It’s easy to install with the Web Platform Installer.
* You can set up and populate databases without leaving your development tool.
* You can bind your user interface to data very rapidly and flexibly.
* With Visual Studio 2008 Professional, you get a debugger for Transact SQL stored procedures.
* You don’t need to learn Visual Basic or Visual C#; PHP can access SQL Server Express databases very easily.
* You get a range of tools for deploying the database to a hosting provider.

This is pretty good functionality for free software.

A broader series of 5-10 minute “How Do I” trainings (Level 101) is available here:

<http://www.msdev.com/Directory/SeriesDescription.aspx?CourseId=124>

To find out more and keep up-to-date with the latest information, bookmark the SQL Server Express Weblog:

<http://blogs.msdn.com/sqlexpress/>

There’s also more information and lots of resources on the SQL Server 2008 Express product information pages:

<http://www.microsoft.com/sqlserver/2008/en/us/express.aspx>

You can find out more about Microsoft’s Web Platform here:

<http://www.microsoft.com/web>

Finally, you can download the Web Platform Installer here:

<http://www.microsoft.com/web/downloads/platform.aspx>