BUILD CLOUD APPLICATIONS WITHOUT HAND-CODING

QUICKLY BUILD AND DEPLOY SOFTWARE-AS-A-SERVICE APPLICATIONS

"Every minute spent on infrastructure programming is a wasted minute."

-Juval Lowy, .NET Software Legend

Increasingly, developers, CTOs, IT business analysts are turning to a new, breakthrough approach for rapidly developing robust cloud applications: Application Generation.

Iron Speed Designer builds database and reporting applications for software-as-a-service cloud environments – without hand-coding. Simply point to an existing database and let Iron Speed Designer generate a visually stunning, feature-rich SaaS application that is easy to customize and ready to deploy.

White Paper
March 2010

Table of Contents

Build Cloud Applications without Hand-Coding

Software-as-a-service (SaaS) applications provide an opportunity to make your business accessible to anyone, day or night worldwide, using a standard Internet connection – all by hosting your application suite in the cloud.

The reasons to do this are compelling: You don't have to pay for any up front deployment infrastructure and you don't have to operate a 24 X 7 server farm environment with all the personnel, equipment and connectivity issues. The cloud computing vendor handles everything for you.

Still, you are responsible for building custom applications that automate your business processes -- the processes that encapsulate your unique way of doing business. Regardless of who hosts your applications or where they are hosted, you still have to build the applications.

Cloud applications differ from regular applications because they are designed from the start with cloud deployment in mind. In particular, cloud applications must be:

- Easy to modify. Since cloud applications express your custom business processes, they must be flexibly designed so they can be quickly updated and re-deployed as your business evolves. Without this, you've sacrificed a key benefit of cloud deployment.

- Easy to deploy. Frequently changing business processes demand frequent no-hassle deployment. Cloud applications must take a minimum of steps to deploy and the deployment must be reliable, requiring a minimum of follow-on deployment verification.

- Highly secure. Your application data and code must be highly secure if they live in the cloud. The power and flexibility of cloud deployment potentially exposes your application and data to unauthorized viewers. You can't rely on your corporate firewall to protect you; instead, the application itself must have rock-solid, world-class *application-level* security.

An application generation tool should generate:

➤ Enterprise-class 3-tier architecture

➤ User interface web pages, including sophisticated data grids with advanced features like multi-table joins, reports, and filters.

➤ User interface code and database bindings

➤ SQL statements & stored procedures

➤ Transaction management code

➤ Workflow reporting

➤ End user account management and access control

➤ Multiple layers of application security

➤ Scalable applications

➤ Easily extensible class hierarchies

Smart IT departments are efficiently conquering these critical issues and freeing up resources using a new breed of software tools called 'application generators' that quickly build and deploy cloud applications.

Application generation takes a giant leap forward, offering a broader, more comprehensive solution, and some tools such as Iron Speed Designer, create your entire cloud application for you without hand-coding.

Accelerating the development cycle without sacrificing performance, features and interoperability, are key benefits provided by application generation.

This white paper describes how one application generator, Iron Speed Designer, builds and deploys software-as-a-service applications without hand-coding. It combines point-and-shoot application generation with Web 2.0 functionality, speeding development and reducing costs.

Business Benefits of Cloud Application Generation

Why is application generation so important and useful for SaaS application development? Because it allows you to:

➢ **Quickly implement SaaS**. Application generators can quickly generate software-as-a-service applications in minutes, with little or no special knowledge. Reducing development time and widening the circle of application developers accelerates your business.

➢ **Empower your distributed workforce**. Cloud applications are available 24 X 7 from nearly anywhere on the planet with an Internet connection. Distributed workforces are easily accommodated without any special infrastructure or setup. Iron Speed Designer, for example, provides built-in deployment tools, expediting cloud deployment.

➢ **Automate business processes**. Custom workflow processes that streamline your business can be quickly designed, built and deployed. Workflow design is often an iterative process, taking several cycles to understand an existing business process and then reorganize and streamline it. Application generation lets you quickly refine screens and workflows, engaging application users in a more iterative development process.

➢ **Move legacy applications to the cloud**. Every business has older custom desktop applications it would like to update. Application generation provides an opportunity to rebuild these applications in a contemporary format and deploy them into the cloud.

Development Team Benefits of Cloud Application Generation

Equally important to the business benefits of application generation is its ability to embrace the work style and culture of your software development team.

➢ **Create cloud applications without hand-coding.** Application generation saves development time, letting you deliver applications faster. Iron Speed's customers report saving an average of 19 weeks per application. No hand-coding is required, unless you're making very specialized code customizations, allowing non-developers to build cloud applications.

➢ **Maintain one single installation**. Cloud applications are easy to maintain because they are usually deployed to a single web farm. Application roll out is easy and uncomplicated.

➢ **Deploy to any .NET web farm in the cloud**. Iron Speed Designer generates standard ASP.NET Web applications that can be deployed to any web farm running a standard .NET framework environment.

➢ **Simplify application maintenance**. Generated applications follow a highly consistent architecture, allowing any developer to maintain any application. There is little or no 'ramp up' time necessary for one developer to understand another developer's application because the architectural knowledge transfers from one application to the next.

What Types of Cloud Applications can be Generated?

From a functional perspective, most cloud applications are database-driven, meaning they principally interact with one or more underlying databases and are typically used for:

➢ Data entry and management – collecting and editing data from users.

➢ Reporting and tracking – reporting, summarizing and visualizing data.

➢ Workflow and scheduling – basic step-wise business processes.

➢ Business process automation – orchestrating data flow between multiple systems.

While there is no precise definition of a custom software-as-a-service cloud application, for our purpose we will characterize them as bringing desktop

application functionality to database and reporting applications via rich transaction-centric user interfaces.

Master-detail pages are among the advanced SaaS application features generated by Iron Speed Designer.

How are these rich user interfaces implemented? Largely by using technologies such as Ajax (Asynchronous Java and XML) and ASPX for the user interface, and web services and ADO.NET for data source access.

Application generation recognizes that the vast majority of most cloud applications consist of standard components – tables, panels, and controls – that lend themselves to automatic generation.

This cloud-based workflow application lets users process transactions from a single Web page.
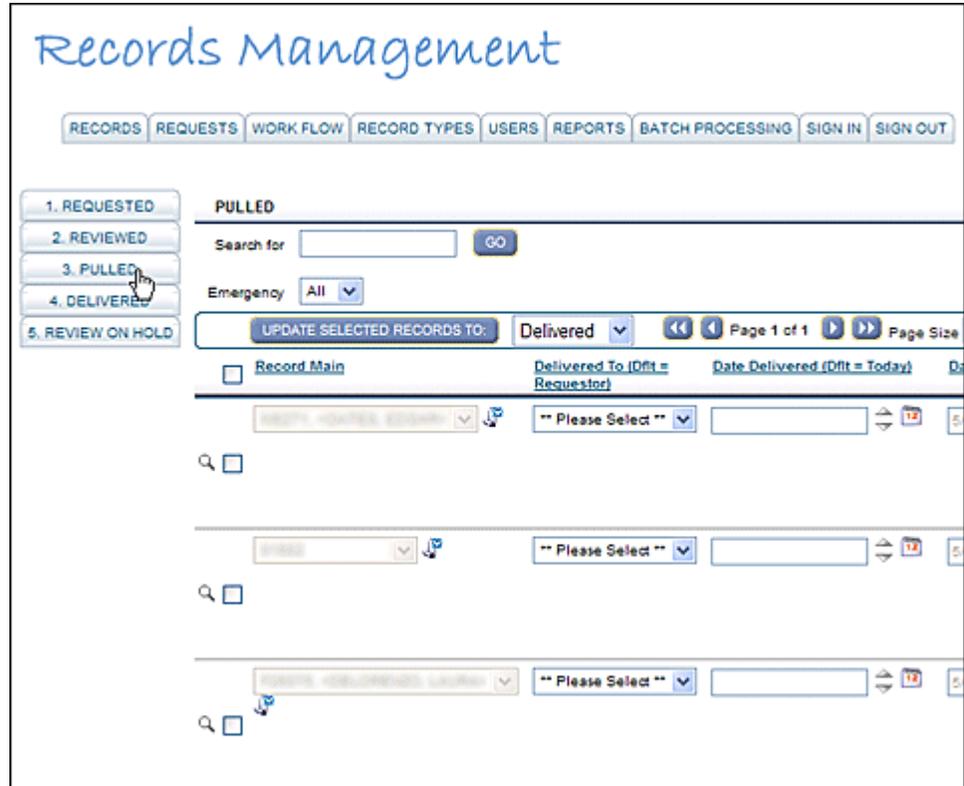
The Iron Speed Designer Approach

Imagine creating a complete cloud application just by pointing a wizard at a database. All the Web pages and components – tables, panels, controls and navigation – are magically created and connected together, and the underlying code and SQL is generated for you. That's exactly what Iron Speed Designer does, showcasing the power of application generation.

A brief example illustrates this point. Suppose you need to build an order entry system that allows sales people to create, edit and retrieve sales orders. This application might have input forms for adding and editing orders and a page for displaying orders for selected customers.

Using Iron Speed Designer, the developer first selects the database tables and views necessary that underpin the application. Next, he selects the page types he wishes to create for the selected tables. When ready, the developer lets Iron Speed Designer generate the desired Web pages, source code and SQL, including code for all of the Web pages, controls, and database connectivity infrastructure.

Database and reporting applications like these lend themselves to application generation because they use straightforward database connectivity and application infrastructure. They all need data entry pages for the selected database tables and views, reporting pages to view and analyze the data in the application, application security, and other infrastructure.

Iron Speed Designer generates complex user interfaces and all the supporting code and SQL, without you having to design pages or write any code. Using Iron Speed Designer, a custom, working application—not just a prototype or individual components—is quickly generated.

Let's take a step by step walkthrough of how Iron Speed Designer generates a software-as-a-service application.

Step 1: Choose a Page Style

The application generation process is simple. Start by clicking 'Create New Application' on the Start tab to launch the Application Wizard. The tool guides you step-by-step through generating your application.

The first step is selecting a page style. Page styles provide a uniform look-and-feel across your entire application without hand-coding each page.
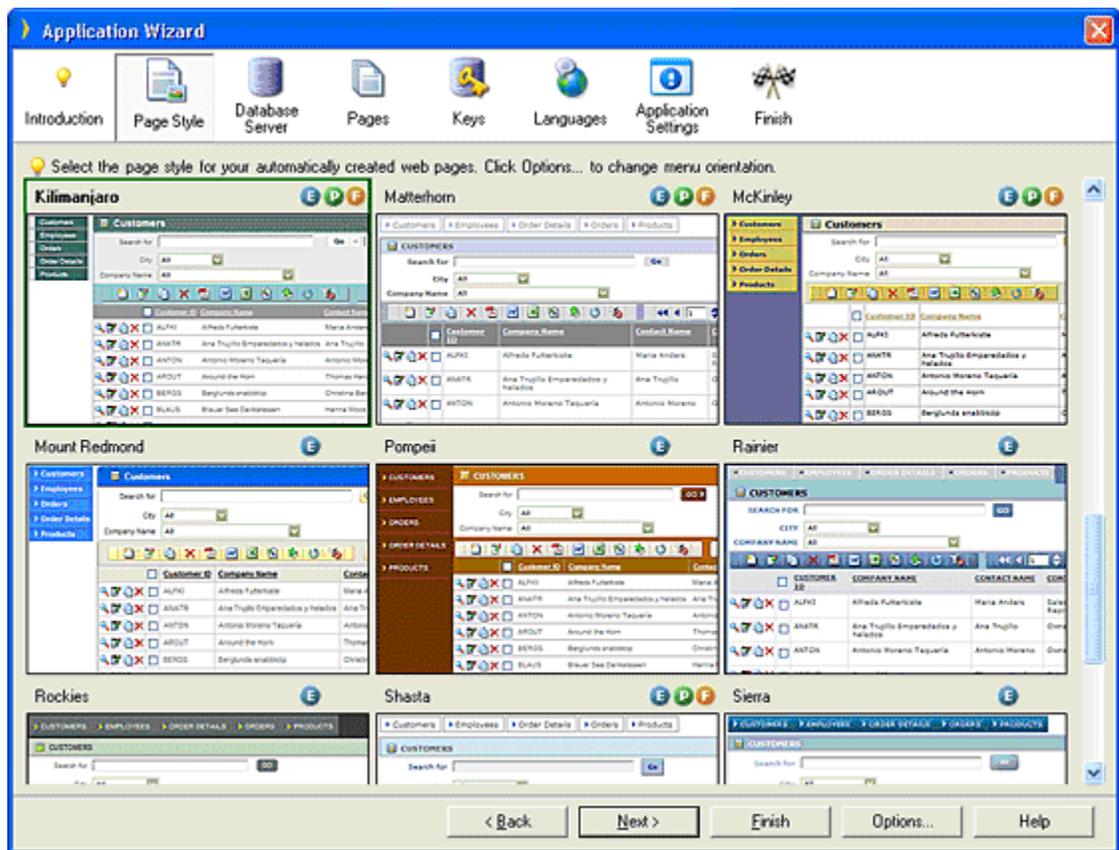
Choose between 27 different page styles to apply a uniform look and feel across your entire application without hand-coding each page.

You can, of course, customize page styles as well as create your own custom styles, a common practice in larger organizations wishing to adopt a look and feel consistent with their corporate colors. One important benefit is the consistency between applications that most users appreciate because it

reduces their overall learning curve; one application looks, feels and operates like the others.

Step 2: Select Database Tables and Views

The second step is selecting the databases you'd like to use in your application. Using your database schema as input, Iron Speed Designer generates data-bound controls, data access and transaction management code, and SQL queries for every page and component that access the database.
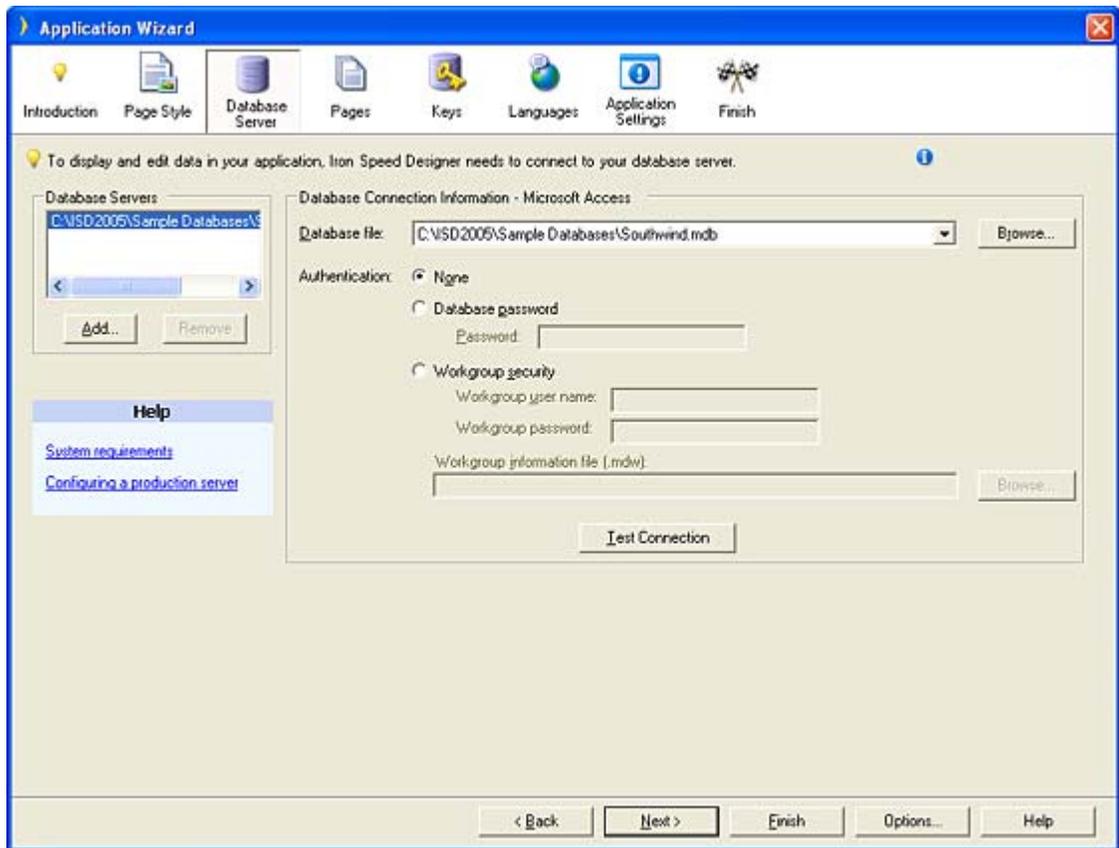
The Application Wizard in Iron Speed Designer generates a complete data access layer for each selected database. Applications may contain multiple databases.

Step 3: Select Web Pages to Generate

Iron Speed Designer automatically creates a suite of pages for each database table and view you select. You can literally create hundreds or even thousands of pages simply by selecting the page types and database tables you want. Think of the time savings compared to building these pages by hand!

Basic page types reflect the basic operations performed with any database table: Create record, Retrieve record, Update record and Delete record – sometimes referred to as *C/R/U/D* pages.

➢ Add Record – a page for inputting data records.

➢ Edit Record – a page for editing data in an individual record.

➢ Show Record – a page for viewing data from one or more queries.

➢ Show Table – a paginated, interactive report view of a query's data.

➢ Edit Table – an editable data grid useful for editing a quantity of records.
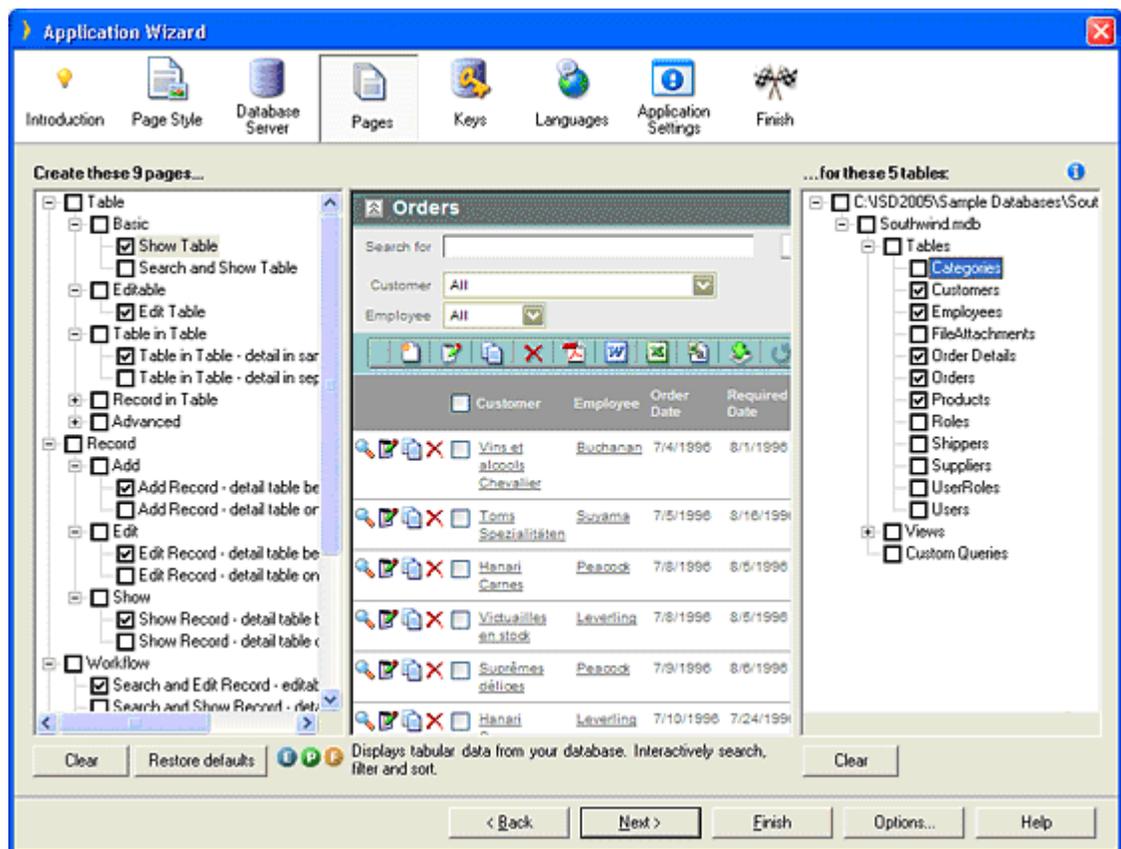
The Application Wizard in Iron Speed Designer generates sophisticated Web 2.0 pages from selected database tables and views.

Iron Speed Designer goes beyond basic C/R/U/D pages, however, by creating a variety of advanced application pages and features you would normally expect to find only in custom desktop applications, such as:

➢ Master-detail pages – all generated pages can be master-detail (parent-child) pages referencing multiple database tables.

➢ Workflow pages – or displaying and processing workflow sequences that automate business processes.

➢ Reporting – pages that summarize data, including PDF and Microsoft Word reports that can be printed and emailed.

➢ Data integration – data import and export in a variety of formats for further analysis by application users.

Step 4: Select Languages and Localization Options

Next, choose the human languages you'd like in your application. Iron Speed Designer supports multilingual applications out-of-the-box and generates a 'select language' application page for your end-users.
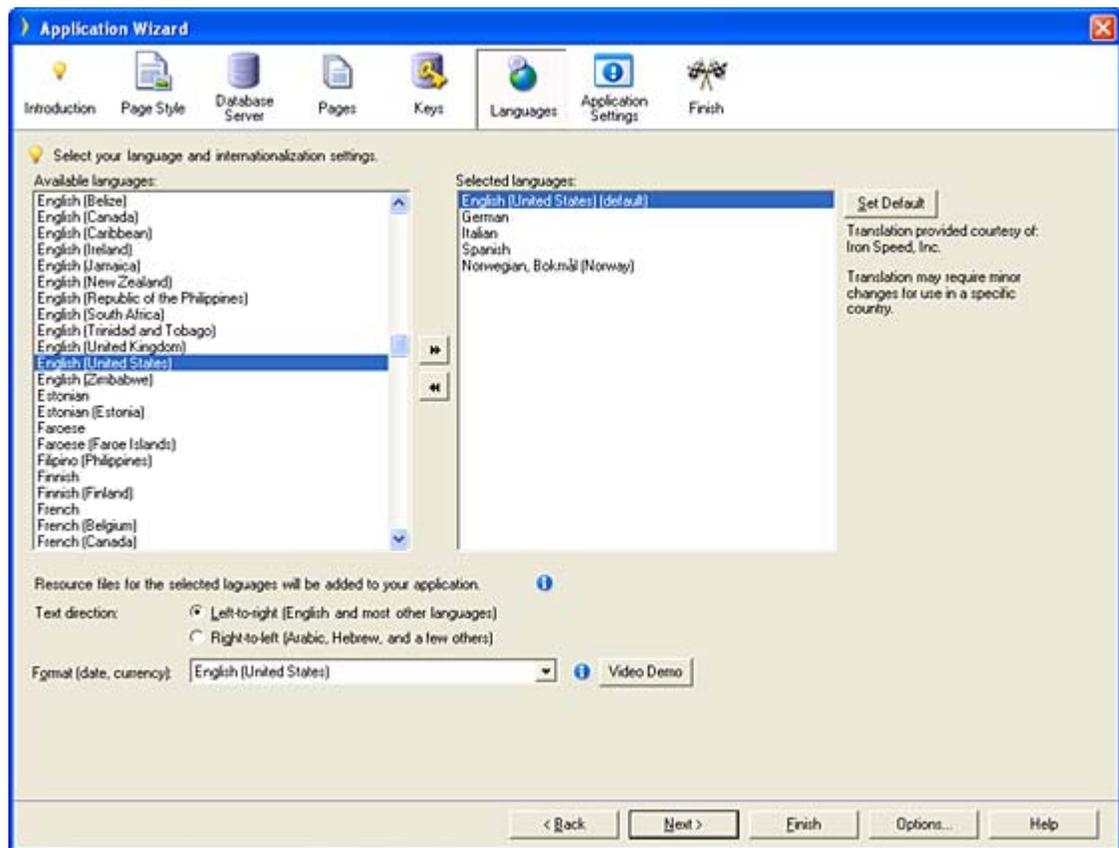

Widely accessible cloud-resident applications frequently require multi-lingual support.

At application run-time, users can switch between languages and the application will immediately display in the selected language. This is very useful for multi-lingual environments, such as Europe.

Step 5: Select General Application Settings

In the final step, select basic application information such as the application's name, code generation language (C# or Visual Basic .NET) and a few other options.
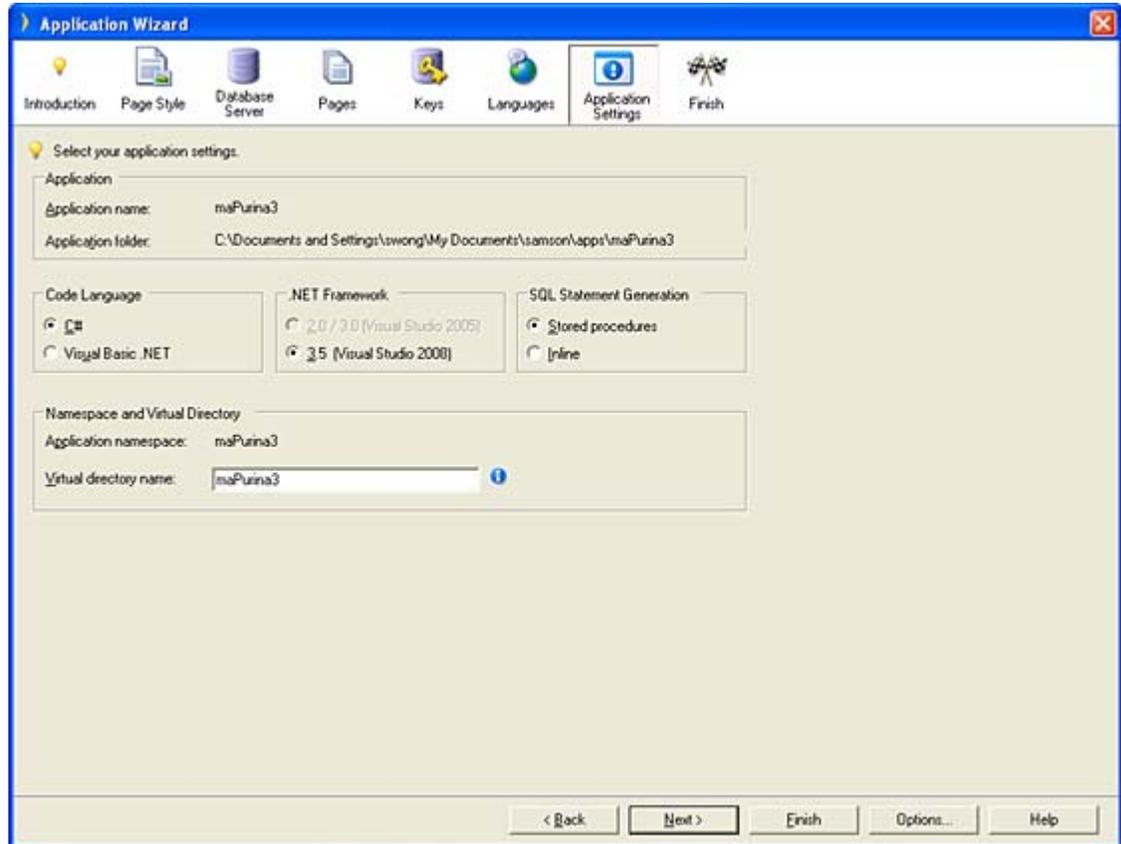
The Application Wizard in Iron Speed Designer generates sophisticated Web 2.0 pages from selected database tables and views.

Step 6: Build and Run Your Cloud Application

Finally, after completing the wizard steps, Iron Speed Designer generates your application using an N-tier architecture that scales easily:

➢ Web pages — the Web pages comprising your application.

➢ Presentation Layer — the user interface code that puts data into and gets data from the Web page controls.

➢ Application or Business Layer — an initially empty layer for adding code customizations, if any.

➢ Data Access Layer — SQL and transaction management code.

➢ Database Layer — database server-resident stored procedures.

All generated pages are automatically linked together with a menu navigation structure, providing an out-of-the-box application with no additional coding required.

These four standard database-connected Web pages are automatically generated for each table in your application.
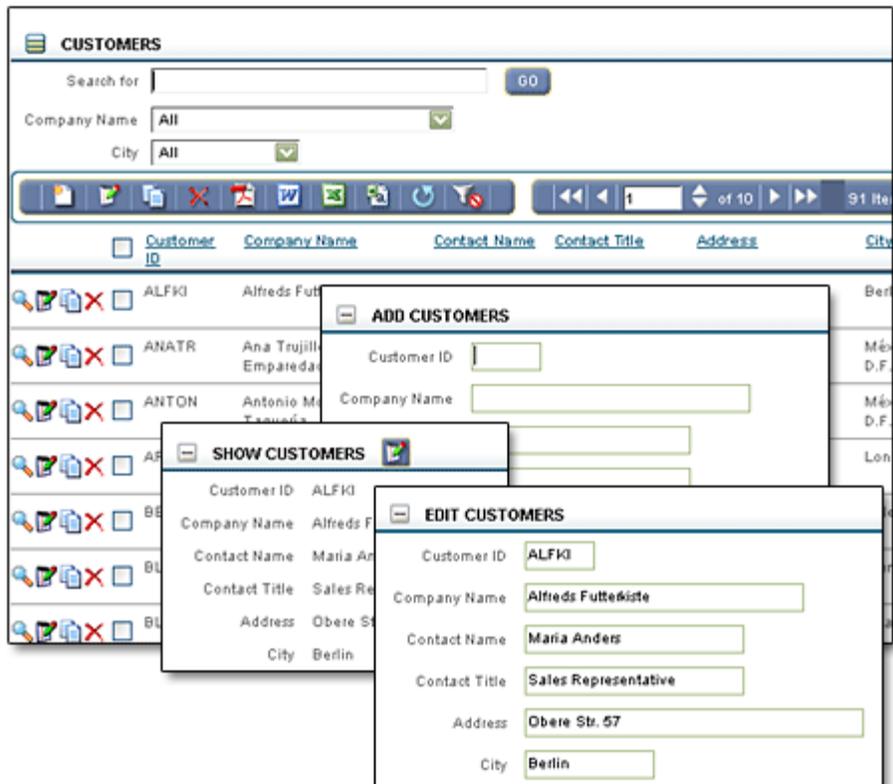
Web 2.0 features extend beyond page types to the components in those pages. These components might include rich text editors, tree controls, multi-level menus and tabbed panels for data input and display. Complex navigation defines Web 2.0 applications with features such as multi-step wizards for data collection and workflow processes governed by application user interaction.

Ajax-enabled controls like pop-ups and rich text editors are examples of functionality frequently found in cloud applications.

Web 2.0 components like this rich text editor generated by Iron Speed Designer bring desktop functionality to the cloud.
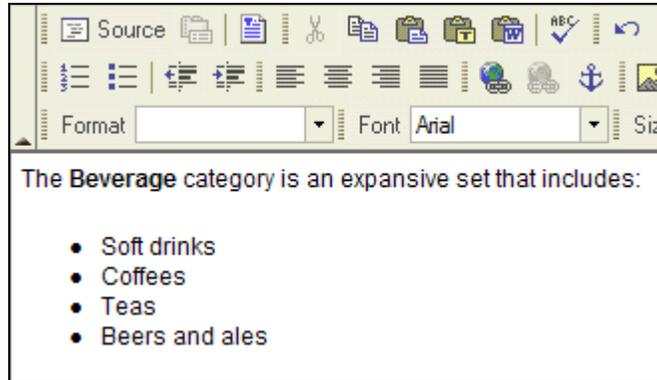
Export formatted data directly to Microsoft Excel for further analysis and reporting.

Advanced application features like integrated security, reporting, multi-lingual support, and data integration are also hallmarks of cloud applications. Features like these are important for applications used by a widely distributed workforce, a central advantage of cloud deployment.

Customizing Generated Applications

It's virtually impossible to generate applications that require no customization – customized page layouts, bespoke application logic, third-party component integration, and interfaces with external systems. While many administrative and data management applications may escape the need for customization, most applications require some amount of customization.

Application generation encourages an iterative development style that includes regeneration that preserves customizations you have made.

Customizing Page Layout

While Iron Speed Designer can make many intelligent assumptions, it can't always create the optimal page layout or even decide the best compliment of fields to include on a page and which ones to omit. So no matter what, you need to customize page layout in a manner that doesn't negate the benefits achieved by application generation.

Iron Speed Designer uses a universally understood spreadsheet metaphor for page layout editing. Individual controls and other components are displayed in spreadsheet cells and pages are rearranged by dragging these data bound controls to their desired locations in the spreadsheet.

An intuitive Excel-like layout spreadsheet makes page customization easy. There is no HTML or ASPX to learn!

When ready, the page is regenerated and viewed in an accompanying 'Live Preview' screen. Iron Speed Designer converts the logical spreadsheet layouts into corresponding HTML table, row and cell constructs in the generated pages.

Since no knowledge of HTML, ASPX or .NET concepts is required to customize page layout, non-developers and para-technicals can build applications as easily as seasoned professionals.

Pages are viewed on a Live Preview screen, displaying actual application data.
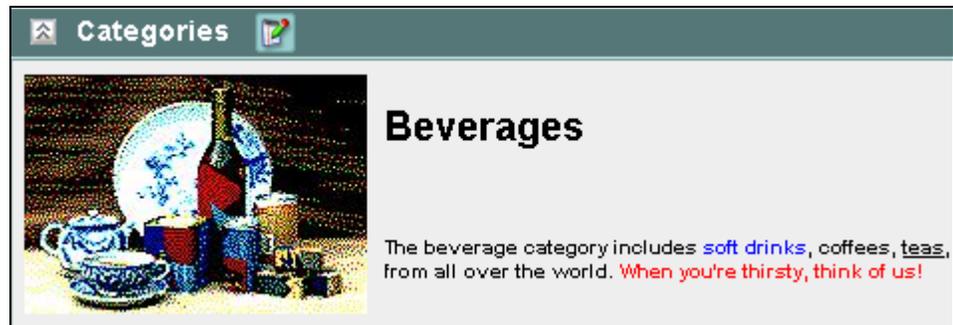
Additional data bound controls are easily added by dragging them from the Toolbox onto the layout spreadsheet. You can, for example, drag complete panels – parent or child, record or table – onto your page, as well as third-party controls and other components. No hand-coding is required.

Drag data bound controls from the Toolbox onto your page. No coding is necessary to connect components.
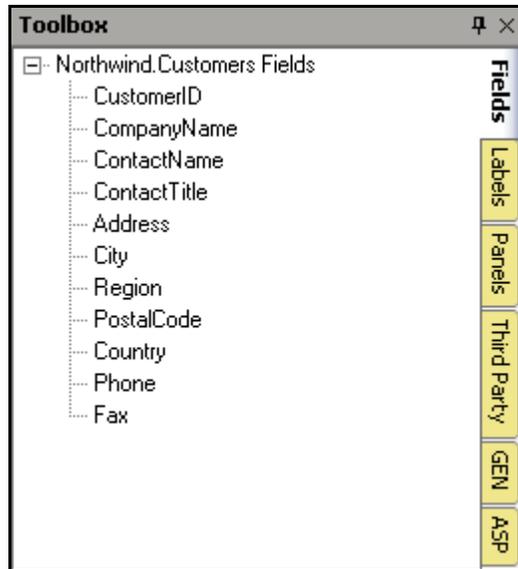
Finally, a 'cell editor' is used for adding any text, custom HTML, ASP.NET tags, or JavaScript to a cell. Iron Speed Designer preserves this custom content and regenerates it as part of the page. The Cell Editor also contains a toolbar for common formatting operations, such as font, font size, boldfacing, text and background colors and cell borders.

Use the Cell Editor to customize individual cell content and display properties.
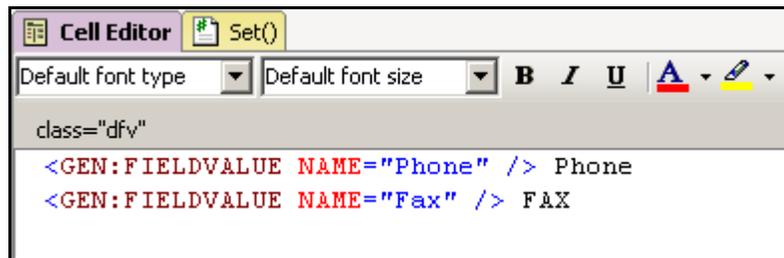
The Cell Editor eliminates the need to understand the overall HTML structure of the Web page. Instead, you only need focus on the modest modifications you make within an individual cell – a very small portion of the page. This greatly reduces the need for the developer to have comprehensive HTML or ASPX knowledge.

Customizing Generated Code

Congratulations, you've generated 100,000 lines of code in 5 minutes! Now where do you place your code customization?

You may want to add custom business logic, add additional user interface components or integrate with external systems. To accommodate this, the generated code must readable, easy-to-understand and easy-to-customize; otherwise the code won't be trusted and the time savings achieved by application generation are lost.

Iron Speed Designer takes an innovative approach called 'code tabs' that combines the page layout spreadsheet with a code editor. Selecting a specific user interface component in the layout spreadsheet displays relevant code methods in a set of code tabs displayed below the layout spreadsheet.

Selecting a page control displays relevant code methods in a set of code tabs.

While this mechanism can't display all possible code methods – there would be way too many code tabs – it does highlight the most commonly customized ones. This focuses your attention on the important code methods, solving the 'forest from the trees' problem of trying to assimilate a very large body of code.

Companion code documentation is generated for each method and is easily accessed by clicking the 'Docs' button in the tool bar within the code tab.

The class hierarchies generated by Iron Speed Designer derive from the base classes in the .NET Framework. Simply subclass the appropriate page, control or database class and add your code customization. The class hierarchy means your code customizations have access to the full functionality of the generated application, all the way down to the .NET framework.
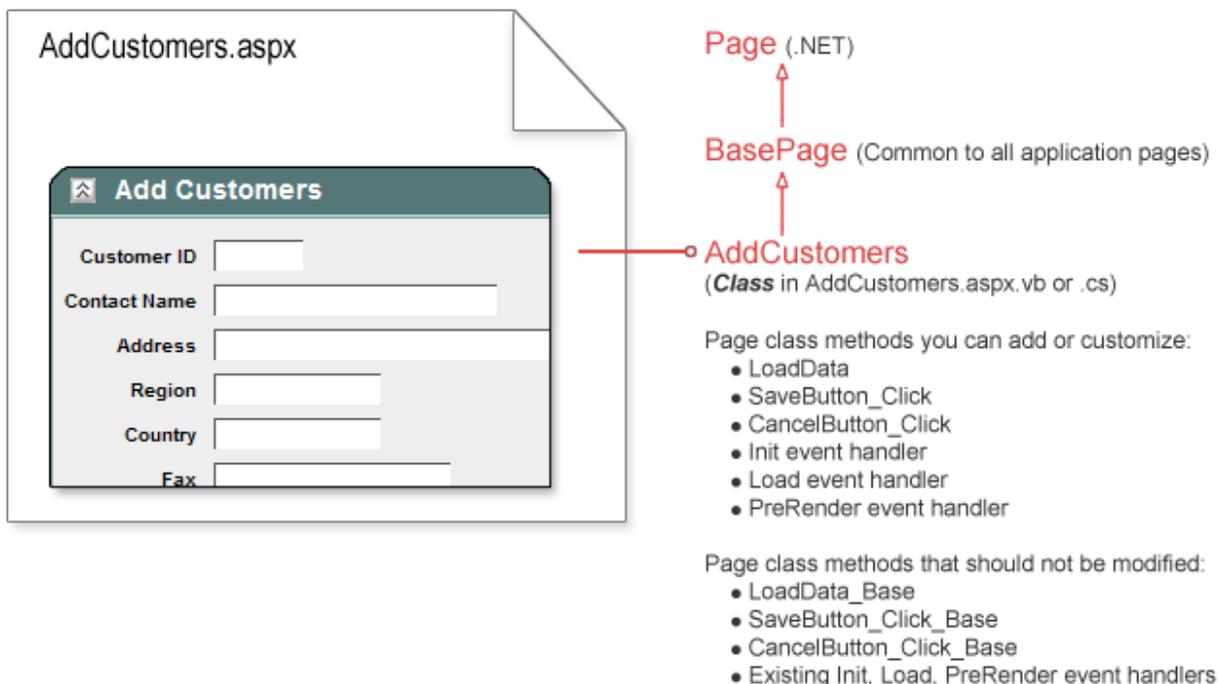
The class hierarchy for a typical Web page generated by Iron Speed Designer.

You can override any of the functions in either the generated application classes or the .NET Framework classes, or write your own methods in combination with the existing base class methods. Then, regenerate the application to reflect your customizations.

Adding Application Security

Most cloud applications serve a variety of constituents both internal and external – customers, marketing, sales, customer service, and management, to name a few. With this wide access comes the need to partition data according to the user type – and sometimes down to the individual user as well.

Although most organizations take security precautions at the network level, this isn't effective for applications hosted in the cloud at an external hosting vendor; application security must be built into the application itself. Here again you can leverage the power of application generation. For example, Iron Speed Designer generates applications with these built-in security features:

➢ Authentication – who can log into an application

➢ Authorization – what can someone do once they're logged in

➢ Component-level (field) security – who can see what field

➢ URL parameter encryption – prevents robotic attacks

➢ Automatic sign-out – guards against unattended machines

➢ Data transmission encryption – prevents unauthorized monitoring

Broadly speaking, Iron Speed Designer generates application security using industry standard protocols such as Active Directory, Windows Authentication, SharePoint groups and Database authentication and authorization.

Automatically generating features like role-based security save many hours of programming.

Basic user authentication governs who can sign into the application. A valid user name and password is required to sign in, giving you control over who can access your application.
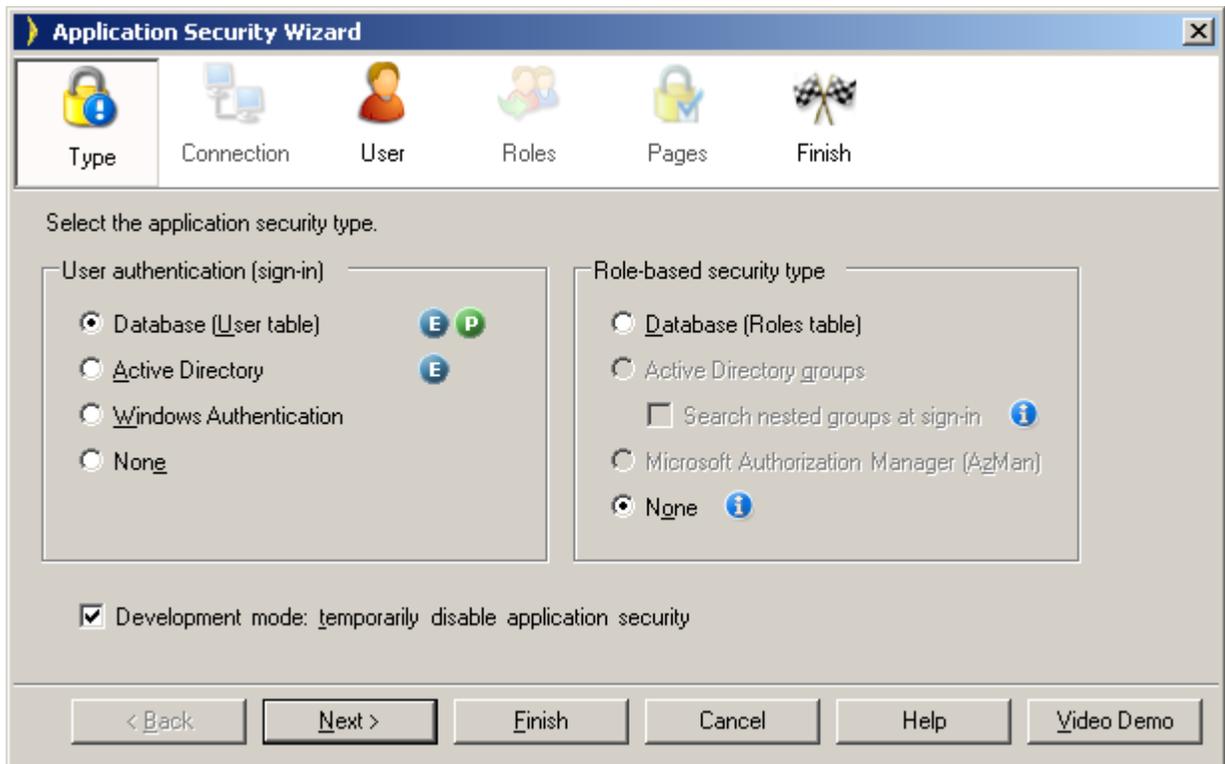
Use the Application Security Wizard to configure authentication and authorization control in your application.

Authorization configures individual Web pages to accept users belonging to one of several designated roles. For example, members of the customer service role may be able to access most customer information but not credit card data. However, members of the "manager" role can access credit card

information and issue refunds or credits. With Iron Speed Designer, users can be assigned multiple roles, effectively giving them tailored access.
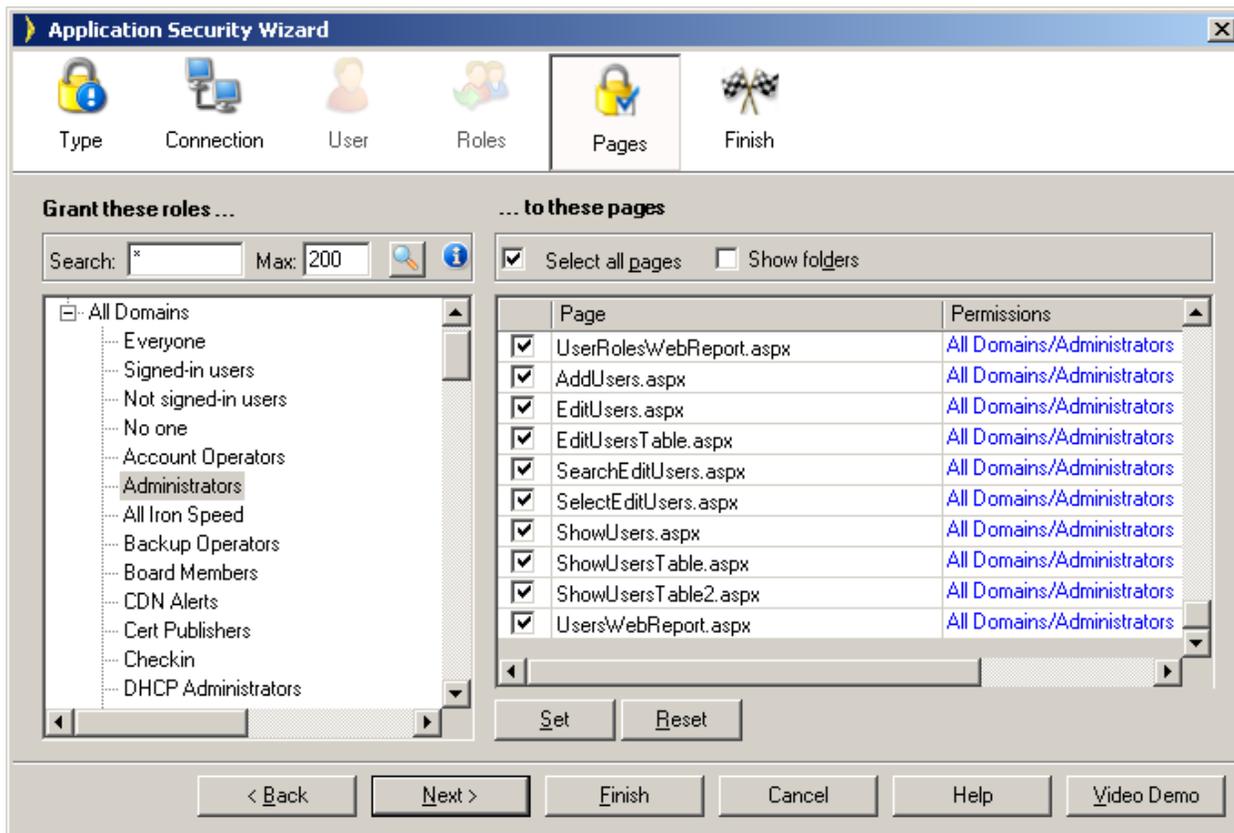
Configure access to specific pages in the Pages step of the Application Security Wizard.

Deploying Cloud Applications

Deploying into the cloud usually means deploying into an unfamiliar environment, an environment you don't fully control because it is owned and operated by a third-party hosting service provider. Moreover, you may be physically sharing servers with many other applications and owners you don't know. To prevent conflicts, hosting service providers tend to operate very unforgiving application environments for the protection of all parties.

So what makes a generated application easy to deploy into a stringent cloud environment?

The application uses industry standard frameworks. Iron Speed Designer generates standard .NET Web applications based on industry-standard databases, such as Microsoft SQL Server, Oracle and MySQL. There are no proprietary libraries or run-time modules present in the generated applications, making them very portable and easy to deploy.

The application is installed into an industry standard production environment. Applications are easy to deploy if they are deployed into industry standard environments such as a Microsoft .NET environment. Proprietary frameworks and platforms make your application non-portable, unable to be moved to another vendor without significant modification.

The production environment uses standard configuration settings. Most cloud computing providers offer a Microsoft .NET environment configured to run in "medium trust" mode, generally meaning your application cannot access any part of the file system, except its own application folder, plus its database and the network. Iron Speed Designer, for example, generates .NET Web applications that are fully compliant with the rigid medium trust mode.

The application can be installed by systems administrators. In some cloud environments, your application is installed and configured by someone who did not develop the application and doesn't even understand its function or particulars. Your application must be easy to install and require minimal or no configuration. Iron Speed Designer, for example, generates a simple, one-step MSI installer, making applications very easy to install into a cloud hosting environment.

Deploying Iron Speed Designer Applications

Iron Speed Designer automatically runs applications on the same machine running Iron Speed Designer, typically a development or test environment. When you are ready to move the tested application into a production environment, you have several deployment options available:

➢ MSI Installer. An installation program is generated which you then run on the production machine to install the application.
➢ Deployment project. A DLL is created containing your application which you then copy to the production machine.
➢ Manual deployment. You copy the appropriate run-time files to your production machine.

Generating an MSI installer for your application is by far the easiest and best way to deploy your application into a production cloud environment. MSI installers are self-contained "executable" installation programs you copy and run on your production machine to install your application. The MSI installer generated by Iron Speed Designer bundles all necessary application files except your database into a single MSI file.

When run, the MSI installation program:

➢ Installs the application files into the appropriate folders.
➢ Creates a virtual directory for the application.
➢ Installs database stored procedures, if applicable.

Generating a deployment project creates a single DLL with your application's ASPX pages, ASCX controls, binary object files, images, and other components. You then manually copy the DLL to your production server for deployment. With this deployment method, however, you are responsible for

creating a virtual directory for your application and installing any database stored procedures.

Deployment projects give you more customized control over your installation, though they require more knowledge and experience. They are best for environments where your IT systems administrator does not allow MSI installers (or other setup programs) to be run or environments where an installation program does not have permission to create a virtual directory or install stored procedures. Deployment projects are also useful when your application is a module in a larger system – hence the DLL format – and is not a stand-alone application.

Manual deployment is best for truly custom environments where you need to control exactly which files are placed where. This method is not recommended for novices but is possible with Iron Speed Designer.

Configuring Iron Speed Designer Applications

After installing your application on your production machine, you must configure your application to run in your production environment.

Step 1: Update your application's database connection strings for your production environment.

Step 2: Create a virtual directory for your application on your production machine.

Step 3: Add your deployed application to the registry on your production machine.

Step 4: Start the session state service on your production machine.

Contact Iron Speed

Building cloud applications can be easy! Let us show you how Iron Speed Designer can accelerate your Cloud application development by eliminating hand-coding. The ROI delivered by Iron Speed Designer means more efficient utilization of key personnel, reduced development and testing time, and more rapid deployment of robust applications.

Telephone: 1-408-228-3400

Web: http://www.ironspeed.com

Iron Speed, Inc.
2870 Zanker Road
Suite 210
San Jose, CA 9514

Please send comments, suggestions and ideas regarding this white paper to editor@ironspeed.com