

Chapter 1 : Migrating Visual Basic and Web Applications to .NET Platform

Most IT Managers prefer to have the full application features available to them and to have their Visual Basic application updated to the next generation of web or modern (WFP) desktop applications. Microsoft's own Visual Basic Programmer's Journal recommends, in order to migrate calendrierdelascience.com "you should rewrite an app rather than port it to.

You MUST have clear in your mind what the actual goals of moving and replacing the current platform and software are. In this article, Joel warns and gives caution against out of the blue simply rewriting perfectly good software that does not rust or wear out. They did it by making the single worst strategic mistake that any software company can make: They decided to rewrite the code from scratch. And this is especially more so, if the new systems not going to accomplish anything more than what the previous system was doing. And, I also done migration of desktop database systems to large mainframe systems. I can only say that it is rare to see an application with that many tables. In fact this issue raises alarm bells right off the bat. If this is not the case, then of course rewriting in. The fact that the data is already in SQL server helps, but that just might mean that you had the horsepower and capacity and infrastructure to scale something that was poorly designed and the first place A very big major portion of software flexibility comes from having properly normalized data models. Unfortunately this put you in a bit of a rock and hard place, because you want to continue to use existing data, and start rewriting functionality in. However rewriting of functionality in. In an ironic twist of fate, this is a catch 22, because likely if that system had really fantastic great designed data models, you might not even have the need to redesign and move this into. And, access supports the use of class objects, and even source code control. In other words keep in mind that people might be asking to rewrite this in. In fact often the opposite occurs, because access is a very RAD tool. This means that the frontline people can often make modifications due to business rules changing, faster than the IT Dept and their team of developers working away on the next great version of the application. I mean, are you to now hire the IT department to build every single spreadsheet and excel for the people because are current business processes are not flexible enough? So in addition to taking access away from these people, you might as well take excel away from them also. I am just pointing out that my spider sense suggests to me that the data models here are going to be a real challenge. Remember, I would always take poor code and great designed data models over the reverse Great code, but terrible data models. The reason for this is with great data designs, then the code and applications practically write themselves. And with great data models, then the ease of which you can change for the ever changing business rules again favors great data designs over that of great code. So, with good data models you can move forms and functionality and the UI over into. So, today we can now allow customers to manipulate and use some of that information that is currently locked up in the system. This might be simple as them checking the status of their orders instead of wasting valuable customer phone time. Or might be something as simple as allowing the customer to look up their account balance. So right now in the marketplace, these self serve customer web portal systems allow customers to enter and use and get at their information instead of the calling up some employed within the organization who then turns around and launches the application and then manipulates the information for that customer while on the phone. Might just as well let the customer do this work! So from order status, to balances owing, to banking, or what ever it is, the real cherry model ticket today is to allow the customer to use at a cell serve web portal that represents all that valuable information that that internal application is creating. As mentioned, you have to ask, is where is the Manpower and personnel going to come to build and maintain this application? Obviously the existing system with enormous numbers of forms and tables you are throwing out must somehow been created, and represents some significant investments of time and efforts. The key concept you have to ask, is where were those significant investments and resources coming from to build that existing application? Who is going to maintain the new system then? In other words you need to design the new system to reduce maintenance costs. At the end of the day, good

DOWNLOAD PDF MIGRATING WEB APPLICATIONS TO VISUAL BASIC.NET

software development and good designs are good designs. Using Access, or VB6, or vb. I should also point out that the new version of access , can create. They are XAML zammel. I am pointing this out, because changing the front end skin from access to. Simply repaint the font ends with. You have some great advice here already. Keep in mind this really comes down to what are the goals and reasons for these people desiring this software to be rewritten in. Those new goals and desires better not be based on the pretext of simply remaking the forms you have now into.

DOWNLOAD PDF MIGRATING WEB APPLICATIONS TO VISUAL BASIC.NET

Chapter 2 : calendrierdelascience.com - Migrating application from Microsoft Access to VB or C#.NET - Sta

I have an application my partner wrote that would allow an autoresponse to any Mapi compliant email that apparently calendrierdelascience.com won't, can someone assist me with.

What Visual Basic 6. When converting from Visual Basic 6. NET, are my ActiveX documents supported? No, Visual Basic 6. You can still interoperate with ActiveX documents from your Visual Basic. What upgrade options do I have when dealing with applications that use ActiveX documents? You can use ActiveX documents from your Visual Basic. You can navigate from a Visual Basic. You can rewrite your ActiveX documents as Web user controls. You will have to include the new controls in a Web Form to simulate the behavior of your ActiveX documents. You can create a Windows Form control that corresponds to the original ActiveX document and then host it in Microsoft Internet Explorer. This approach takes advantage of the similar features in both the source and target components, but nonetheless is limited with respect to deployment and compatibility. How are Visual Basic 6. In Visual Basic 6. When a Visual Basic 6. NET Web application project. Any code related to state management will need to be replaced; there are many options for this. Some options involve keeping information on the client for example, directly in the page or in a cookie , and others involve storing information on the server between round trips. In a VB to. NET conversion, what techniques can be used if I have decided to keep the information on the client? If you decide to store your instance on the client, you can use some of the following techniques: ViewState property provides a dictionary for retaining values between multiple requests for the same page. This information is automatically stored. When the page is processed, the current state of the page and controls is hashed into a string and saved in the page as a hidden field. When the page is posted back to the server, the page parses the view state string at page initialization and restores property information on the page. You can create hidden fields on a form that are not visibly rendered by the browser. With this technique, you can set properties just as you can with a standard control. When the page is submitted to the server, the content of the hidden field is sent in the HTTP Form collection along with the values of the other controls, which allows you to store information directly in the page by way of the hidden fields. You can use cookies to store information about a particular client, session, or application. The cookies are saved on the client device, and when the browser requests a page, it sends the information in the cookie along with the requested information. The server can read the cookie and extract the necessary value. You use query strings to maintain some state information, but they are limited to the capacity of the browser and client devices. This imposes a character limit on the length of the URL. And if I decide to store the information on the server, what methods can be used? NET, you can save values using application state an instance of the `HttpApplicationState` class for each active Web application. Application state is a global storage mechanism accessible from all pages in the Web application and is useful for storing information that needs to be maintained between server round trips and between pages. Application state is a key-value dictionary structure created during each request to a specific URL. You can add your information to this structure to store it between page requests. NET, you can save values using session state an instance of the `HttpSessionState` class for each active Web application session. You can maintain the state of the page using database technology when you are storing a large amount of information. Database storage is particularly useful for maintaining long-term state or state that must be preserved even if the server must be restarted. The database approach is often used in conjunction with cookies. These event procedures will be upgraded by the upgrade wizard, but they will not be called at run time. After upgrading, you will need to move any code in these events to equivalent ASP. NET events, such as `Init` or `Unload`.

Chapter 3 : Migrating a Delphi 7 application to .NET - Stack Overflow

Applications that contain business logic at the form layer will not easily migrate to Visual calendrierdelascience.com Move the logic into components and classes for a smoother transition. 2.

NET in 9 months. There are 5 basic options dealing with a legacy VB6 application: Leave it as it is â€” if the application is rarely updated or changed Replace â€” if there is a similar commercial or open source application or service Migrate to .NET â€” using a migration tool which converts all VB6 code into equivalent .NET code Extend with .NET â€” add new functionality using .NET â€” hand writing the entire code in. This is a viable approach when the initial VB6 application was poorly written and a re-write is desired, the application needs a major change to satisfy new needs, or the source code is not available. If migration is the path to go, the next three solutions are helpful with that: Microsoft Visual Basic Upgrade Wizard. It is included in VS and works well with small and medium projects. For large, enterprise applications, the next two solutions are recommended by Microsoft. Visual Basic Migration Partner by Code Architects Microsoft recently published a case study highlighting the success of the migration of , lines of VB6 code to .NET. The project was an ERP system built over 10 years and consisting of 33 applications. The best solution for this application was to migrate it rather than use a customized ERP application or rewrite it. Customizing a commercial ERP was not accepted because it would have cost million Euros, it would take 2 years and no vendor guaranteed the implementation of all original features. SiS selected 25, lines of code and tried to migrate them with various tools. VB Migration Partner was the best, the chunk of code being migrated, compiled and ready to run in 2. They also reported that VB Migration Partner required the least amount of manual intervention to make the UI look like the original one. The process evolved in phases: Companies with major investments in VB6 can still make good use of their legacy assets by migrating them to .NET. Microsoft and its partners seem to have the tools for the job.

Chapter 4 : Migrating from PHP to calendrierdelascience.com

calendrierdelascience.com Web pages, to an calendrierdelascience.com Web application, are what individual Forms are to a Visual Basic application. That is, an calendrierdelascience.com Web page represents an interface with which the Web visitor interacts.

Migrating Visual Basic Applications to the .NET platform has been under serious discussion ever since Microsoft launched its newest series of products under much hyped vision statements. .NET is one of the key products that enable application development under the new vision. .NET is not quite backward compatible with prior versions like Visual Basic version 6. This makes migration a serious issue. This article looks at possible reasons that one would choose to migrate, and outlines certain important issues to be considered when such a migration is attempted. Why Migrate to VB. The first question that strikes ones mind is "why should I migrate? This is what makes Java applications robust when compared to older generation Microsoft applications. CLR manages the code in a very similar fashion to the Java model, thereby making applications much more Robust, Stable and secure. The application also becomes more maintainable because of the managed code. .NET applications are packaged as self-contained, self-describing, versioned assemblies collection of classes and meta data. One can deploy, remove and manage N different versions of an application or DLL on the same machine without creating conflicts. This enables a product company to easily manage versions and upgrades without causing the headaches so common with usual upgrades or OEM-versioning. Desktop applications can be much more easily converted to Web Based applications using the WebForms paradigm of .NET made available through VB. XML Web Services creation and integration as facilitated by the .NET platform are readily available through VB. This makes the creation of hosted applications easier, thereby providing wider access to proprietary intellectual property. XML and web services also enable Enterprise Application Integration through standard methodologies as available through the Microsoft family of. Microsoft strategy is to eventually migrate all its products to the .NET products would be obsolete. .NET applications when developed carefully can be platform independent. Managed code built using .NET will run on platforms that host the .NET framework irrespective of the Operating System or machine type. Such non-Microsoft platforms have started to appear already. If one is convinced of a necessity to migrate to VB. .NET, the next question that arises is "How to migrate? .NET and then see the various strategies for migration. .NET implements a new paradigm and has many advantages over VB6, few of which are listed below. All features of the .NET framework are readily available in VB. .NET is totally object oriented. .NET , which follows the disconnected paradigm , i. It also retrieves the records that are expected to be accessed in the immediate future. This enhances Scalability the of the application to a great extent. Error handling has changed in VB. This again credits to the maintainability of the code. Another great feature added to VB. .NET is free threading against the VB single-threaded apartment feature. In many situations developers need spawning of a new thread to run as a background process and increase the usability of the application. .NET allows developers to spawn threads wherever they feel like, hence giving freedom and better control on the application. Security has become more robust in VB. In addition to the role-based security in VB6, VB. .NET comes with a new security model, Code Access security. This security controls on what the code can access. For example you can set the security to a component such that the component cannot access the database. This type of security is important because it allows building components that can be trusted to various degrees. The CLR takes care of garbage collection i. This relieves the developer from thinking of ways to manage memory. CLR does this for them. Methods of Migrating After getting a feel of VB. The various methods that can be used to migrate the application are covered below. In many cases, these approaches will be combined. This wizard does almost the entire migration barring some modifications that must be made to complete the upgrade process. Fresh Design Redesign of certain core modules or perhaps the whole application is a possible approach if new features of the .NET architectural framework are considered. Managed code under CLR can

run in tandem with unmanaged code e. Therefore it is possible to reuse many existing components as such. Use of Customized tools to enhance migration Porting driven by a wizard and re-design are easier when migrated using customized tools like the "Cal-Migrater" developed internally by Calsoft Labs for a health care product migration. This tool makes use of both the wizard and redesign approach to make faster pace through the. As mentioned earlier, porting the application using the wizard approach does automate the process to an extent but some modifications must be made to complete the upgrade process. The wizard produces a report in HTML format with the information of the places where it failed to migrate the application to. There are number of changes that can be done to the VB6 application to make the upgrade process much more efficient. Some important aspects are listed below. Avoid using late binding. This is because properties and methods cannot be verified during the upgrade process. VB6 specifies a default property for every component. For example, the default of Textbox is Text property and Caption is default for Label component. NET there is nothing called as default property. In VB we can declare an array with any positive integer as its lower bound. NET all the arrays are zero bound. Lines and Shapes are not supported in VB. NET and hence cannot be upgraded. Instead a graphic object is provided for shapes. Use constants instead of underlying values. VB6 constants will convert to the correct value when upgraded to VB. NET, but if actual values are used then it may end up with wrong hard-coded values. For example, for Boolean values using -1 and 0 instead of True and False and will have an adverse effect in VB. NET True is 1. Re-Design of the whole application can be considered if the application will Grow over time. NET being object oriented, the code can be reused and the application can be extended with less effort and time. Be taken online, i. NET provides web forms that can be used both for web application and VB application. This gives it dual manageability. Need free threading modules which perform background tasks. Migration Strategies There are two strategies to migrate using the above methods viz. Vertical migration and Horizontal migration. Both these strategies help us find out key areas that need an immediate upgrade and the modules that can be upgraded later. Vertical Migration In vertical migration, a module of the application that is fairly standalone or isolated is taken, and all the tiers of this module are migrated to VB. Thus migrating one module will give a good idea on the effort required to migrate other modules. Horizontal Migration In horizontal migration, one layer of all the modules is migrated to VB. NET followed by other layers. For example, first the UI would be migrated then the middle tier. Horizontal migration is ideal in situations where the modules are tightly coupled and the effort and risk involved in the migration is known to a large extent. Conclusion Microsoft is migrating all its technologies into the. There are several reasons and advantages for migrating existing applications to. This article has taken a summarized look at some of the benefits and issues associated with such a migration.

Developing Web Applications with Visual calendrierdelascience.com and calendrierdelascience.com John Alexander and bestselling author Billy Hollis show programmers how to develop enterprise-level Web applications using Microsoft's popular programming languageâ€™ Visual calendrierdelascience.com

At the time of this writing, the .NET Framework is at version 1. The .NET Framework Version 1. The .NET Framework contains the framework of classes needed to execute .NET applications on the Web server. .NET Web applications are .NET applications and utilize the many classes in the .NET Framework, it is essential that the .NET Framework in any order, it is recommended that you install the Web server first. This is because when installing the .NET Framework. More information can be found at 4GuysFromRolla. .NET can make the process much easier. There exist other development editors for ASP. .NET offers a design interface that most closely resembles the design interface for Visual Basic 6. .NET application is to creating a classic Windows desktop application. Start by firing up Visual Studio. .NET Web Application project. .NET application to exist. If you are planning on serving the ASP. If you are hosting your ASP. Figure 2 shows a screenshot of the Visual Studio. .NET Web application at http: Upon clicking the OK button, Visual Studio. .NET will create the new Web application. Each Web page is a distinct file with the .asp extension. That is, an ASP. .NET Web page represents an interface with which the Web visitor interacts. .NET Web page is composed of two distinct parts: A graphical user interface A class whose code responds to events raised by the controls in the user interface Oftentimes ASP. The controls that can be added to an ASP. Upon creating a new ASP. Figure 3 shows a screenshot of Visual Studio. .NET Web application project. .NET Web page, showing you what users visiting the Web page will see. To add Web controls to the ASP. Therefore, you can work on the ASP. To get acquainted with Visual Studio. .NET Web page that functions as a simple financial calculator; one that computes the amount of interest someone will realize on an investment, given a starting principal, an interest rate, and the investment duration. The Label Web controls, like in Visual Basic 6, display either static or dynamic text content. Next, click on the Label Web control to load its properties into the Properties pane. Here you can set its Text property to the value of the text you want to display and its ID property to some unique value. Similarly, you can add a TextBox Web control to the Designer. Figure 4 shows a screenshot of the Designer after all of the needed Label and TextBox Web controls have been added. Note that there is a TextBox Web control for each user input that needs to be collected. When adding these Web controls, you should set the ID property of each Web control to a meaningful value. The first one is a Label Web control that will display the results of the financial calculation. Place this Label Web control near the bottom of the page; clear out its Text property and assign its ID property the value, Results. This Web control will display a button that the user will click upon entering the data. After adding these two Web controls your screen should look similar to the screenshot in Figure 5. Specifically, we want to perform the financial calculation when the Button Web control is clicked and display the results in the Results Label. .NET Web page, which will have the new event handler shown in Figure 6. The source code portion for an ASP. The Click event handler has been automatically added to the ASP. All that remains left to do is to write the code to compute the future value of the investment, and to display this value in the Results Label. The following shows the complete code for the Click event handler. As a Visual Basic developer, this code should be fairly straightforward. Object, ByVal e As System.EventArgs. This will compile the ASP. You should see a browser window similar to Figure 7. .NET applications a more familiar one for Visual Basic developers, there are a few subtle, yet profound differences that must be understood. The most basic difference between Web applications and desktop applications is the medium they are delivered on. Furthermore, the state of the application is self-contained. That is, if the user needs to fill out several forms worth of data, each form can easily know about and access the values from previous forms. With Web applications, however, these same rules do not apply. Take the financial calculator applicationâ€™ when a Web visitor requests the ASP. At this point there is no direct connection between the ASP. From his computer, our

user will enter data into the various textboxes and then click the "Display Future Investment Value" button. NET Web page, sending the value the user entered into the assorted textboxes along with other information. From this information, the ASP. This Click event handler, as we saw earlier, computes the final investment value and displays this value in the Results Label. Realize that the disconnected, stateless nature of the Internet places some restrictions on the functionality Web applications can provide. For example, imagine that you were designing an application that prompted the user for their ZIP code, among other things. With Visual Basic it would be very easy to add a TextBox control and to trap each keystroke made to that control. If, for instance, the user entered a keystroke that was not a number, you could either not accept the input or provide some sort of feedback mechanism to let the user know that they had entered a character that is invalid for a U. With a Web application, however, such functionality is not possible from the ASP. This is because the ASP. This only happens when the page is first loaded, or when the page is submitted back via a button click or some other mechanism. Since Visual Basic desktop applications are connected and stateful, a common Module can be used to hold values common to all Forms, or the master Form could expose the selected value of its drop-down list to the detail Form. Due to the disconnected and stateless nature of the Internet, the only time a Web Form is executed is when it is requested from a Web browser. Redirect method, which instructs a Web browser to request a specified URL. It is through this method that one Web Form can load another Web Form. To show the Response. When this Button is clicked, the user will be sent to a different Web Form. Start by creating the first Web Form, the one with the Button, and name it First. Figure 9 shows a screenshot of Visual Studio. A Button Web control has been added to the First. In this event handler, add the following line of code: Now, create a Web Form called Second. Do so by right clicking on First. This will display a button in the browser. This button, when clicked, will cause the browser to re-request First. Upon being requested, the source code portion of First. This will cause the Response. A set of simple scalar values—like strings, numbers, and dates—can be passed from one Web Form to another via the QueryString. Redirect method call so that the value of the string is passed along to the Second. Finally, in the Second. To start, add a TextBox Web control to First. This can be accomplished by changing the code in the event handler from:

Chapter 6 : Case Study: Migrating a VB6 Large Application to .NET

There were heavy changes in calendrierdelascience.com Windows Forms applications migrate more cleanly but the new partial class form model won't be generated automatically for the converted forms.

One option which is the default with the shipping VS bits is to migrate the project to use the new VS Web Site project model. This model is much less invasive and requires fewer changes. Please follow the below steps in exact order. If you have problems using the below steps, please post in this forum and VS team members will be able to help. You can learn more about it and install the release candidate build from here. After installing it, please spend a few minutes following the tutorials on this site to test it out and learn the basics of how it works. There is the chance that you might need to roll-back to the VS solution if you run into issues, or start over from any step.

Copy Remote Frontpage Projects to your local machine If you use Frontpage Server Extensions to access your project on a remote server, you will need to copy it locally before migration if your web projects run locally then you can skip this step. If you do not copy the project locally first, migration will convert the project to a VS Web Site project instead. To copy the remote site locally use the VS Copy Project menu command under the project menu. This will launch the copy project dialog. Change the destination to "http: After the project has been copied locally you should remove the remote one from the solution and use "Add Existing Project" navigating to the c: Make sure the project still builds in the IDE. Open Your VS Web Project and Make Sure it Compiles Clean Before attempting to migrate anything, please make sure that you open up your existing VS solution and perform a clean-re-compile of the entire solution -- and then run the application. Spending 2 minutes to verify everything is working before you migrate can save a lot of grief later especially if the cause is because of a last-minute directory move, etc. Once converted, you can then add it back under source control. This will cause the VS Migration Wizard dialog to launch: Choose a location to backup the project: The conversion wizard will then perform a few steps: Compiling and Running your Web Project You are now ready to compile your web project. NET now has a built-in menu and preview control as well as built-in profile, membership and rolemanagement APIs -- so if you declare these types in your VS web projects today you may get an "ambiguous type" compiler error when you compile note: You may or may not also run into some API changes when moving to V2. This site lists the known set of breaking API or functionality changes that were made between V1. You should consult this if you run into a problem. Run the Web Application Run the application to verify that the application is working fine. Fix up any runtime issues that you find.

Convert to Partial Classes When you migrate your project using the above steps, none of your code-behind page code or classes are modified. This means that the code should look and work just like it did in VS This makes it much easier to migrate existing code to VS You can optionally choose to keep your code in this format. Doing so will require you to manually update the control field declarations within your code-behind file -- but everything else will work just fine in VS This is done by separating out your current code-behind files into two separate files -- one for your code and event handlers, and the other a. Please review this tutorial for more details on how this new code-behind model works. To migrate your code to use this new model you should follow these two steps: You want to make sure that all code is compiling clean without problems before attempting to update it to use partial classes. Please test this before proceeding. This will then iterate through each page or user control within your project and migrate all control declarations to be declared within the. You should then do a clean re-build of your web solution. If you have any errors in your task list, you can navigate to the appropriate page in the solution explorer and choose the "View Code" and "View CodeGen" context menu items to examine the code and fix any issues. If for some reason the. VS will then avoid re-adding it to the. You might consider doing this if you want to watch closely the changes made on each page.

DOWNLOAD PDF MIGRATING WEB APPLICATIONS TO VISUAL BASIC.NET

Chapter 7 : Web Application Migration, Website Integration and Website Redesign Company

I have an app that I'm re-writing for MySql and MsSql, the old version connected to an Access database. It is more than likely that existing customers will want to upgrade to the later version.

NET, follow these steps: On the File menu, point to New, and then click Project. Under Templates, click ASP. In the Location text box, type WebApplication1. If you are using the local server, you can leave the server name as http: Create project folders on the target computer Create the project folder. To do this, follow these steps: Create the WebApplication1 folder at C: Give Full Control permissions to the Authenticated users. For more information about how to share the files and folders, click the following article number to view the article in the Microsoft Knowledge Base: Click Start, point to Settings, and then click Control Panel. Expand the server name. Locate the content folder that you created to hold the content c: Click Next to continue. Create the solution folder. To do this, create the solution folder WebApplication1 for. This is located at C: Note UserName is the user name to log on to the computer. Copy the project to the target computer To copy the project to a remote computer, follow these steps on the source computer: Connect the folder C: On the Visual Studio. In the Destination project folder text box, type http: WebApplication1 is the virtual directory and is also the name of the folder that you created in step 2 of the "Create Project Folders on the Target Computer" section of this article. In Web access method, click the File Share option. In the Path text box, type F: Note F represents the name of the mapped network drive. In Copy, click to select the All the files in source project folder option. Click OK to copy the project. Note While copying the project, you may receive the following error messages because of incompatibility between the versions of Internet Information Server IIS on the source and the target computers. Microsoft Development Environment The Web was created successfully, but an error occurred when trying to configure the application root for this Web. Web projects may not operate correctly without an application root. The returned error was: Active Directory Services cannot find the web server. A possible cause for this is an incompatibility between versions of Internet Information Server IIS on the client and the server. The usual cause for this error is creating Web projects on a Windows or newer server from a Windows NT 4. You can manually configure the application root using the Internet Services Manager. You can manually restrict browse access to this folder using the Internet Services Manager. Type the address of your virtual server and the project file name for example, type http: On the Build menu, click Build Solution. To save the solution. References For more information, click the following article number to view the article in the Microsoft Knowledge Base:

DOWNLOAD PDF MIGRATING WEB APPLICATIONS TO VISUAL BASIC.NET

Chapter 8 : Migrating a VS Web Project to VS

Upgrading Visual Basic Applications to Visual calendrierdelascience.com and Visual Basic Important! Selecting a language below will dynamically change the complete page content to that language.

Hypertext Preprocessor 4 to ASP. NET, as well as the underlying functionality and architecture of the two systems. NET have several major differences. NET is not a language or a parser but rather a set of technologies in the Microsoft. NET, though, is different in that it provides a robust, object-oriented, event-driven programming model for developing Web pages, while still maintaining the simplicity that PHP developers are accustomed to. This allows for more rigorous OOP features, such as inheritance, encapsulation and reflection. In this paper, we assume that the reader has experience with PHP as well as programming and software development in general. We begin this paper with a look at code with a short comparison of the underlying architectural differences and the OOP development model, followed by a feature comparison, and then a comparison of Syntax and Common tasks for developing Web applications with PHP and ASP. NET are relatively similar with analogous functionality and syntax. NET at a lower architectural level. NET is a framework built upon a series of technologies such as the CLR and offers an extensive series of well-organized class libraries that provide for most every conceivable set of functionality that would be used in a Web application. It also allows for the easy and simple creation of components to extend the framework. For example, PHP only supports partial encapsulation such as support for declaring methods and fields in the class and partial polymorphism no overloading, no abstraction. PHP also lacks support for such concepts and accessibility in that there is no concept of private, public, or protected functions in classes as well as the Overloading. This has both an upside and a downside. The downside is that for some Web developers there is a steeper learning curve for ASP. NET intuitively familiar and easy to learn. The upside to ASP. NET applications for the most part result in better designed code, have clear separation of content, logic, and data and thus are generally easier to support over the long term of a applications life cycle. You can find an introduction to the main areas of object-oriented programming from a Visual Basic point of view in Object-Oriented Programming in Visual Basic. Opcodes are low-level binary instructions that will be used to serve the PHP page. There are a number of commercial products that can be used to speed up the execution of a PHP page by optimizing these opcodes. If this is the first time the page has been requested, ASP. Then the request is run using this compiled code. Subsequent requests are served from this same machine code assuming the page has not been modified. It is important to note that the binary code that is generated by the CLR is already as optimized as possible; no add-on product is necessary to achieve maximum performance. It is also important to note that everything in ASP. NET is compiled to machine code before being run. Even HTML text is converted to a string literal control and inserted in proper order into the control tree.

Chapter 9 : Upgrading Visual Basic Web Applications to .NET

Migrating from VS to VS using the Web Application Project There are a couple of different strategies you can take when migrating a VS Web Project to VS One option (which is the default with the shipping VS bits) is to migrate the project to use the new VS Web Site project model.