Introduction to IntraWeb

About This Document
This document is designed as an introduction to IntraWeb and to provide an overview. For more detailed information please see the website and the documentation that ships with IntraWeb.

This document is also continually being updated with each release. Be sure to check back for updates.

If you have questions that you cannot find answers for, please post a message on our newsgroups and you will receive a prompt reply. Information about our newsgroups is available at http://www.atozedsoftware.com/support.html.

If you have never done web development before, you will love IntraWeb. It is just like building normal Delphi applications. You should read “How IntraWeb is Different” on the website in addition to this document.

What is IntraWeb?
IntraWeb is a framework and component set for building web applications in a true RAD manner. Using IntraWeb you can create, debug and Maintain web-based applications as quickly and easily as your normal Delphi applications using IntraWeb. Want to show a form? Call it's .Show method. Want to show a message? Call WebApplication.ShowMessage. Could it be any easier? Using IntraWeb you can deploy your applications as stand alone services, ISAPI DLLs, or Apache DSOs. No other development tool creates web applications as fast and as easy as IntraWeb!

Take a look at just some of IntraWeb features:

- Supports Delphi 5/6 Pro or Enterprise.
- Pure Delphi, no JavaScript to write!
- TFrame support.
- Includes Built-in HTTP server, makes debugging Easy!
- Netscape 6 and IE 4-6 Browsers supported.
- Netscape 4 Browser supported by IntraWeb 4 only.
- Integrated WYSIWYG HTML Editor.
- No Java or ActiveX to download or install on clients.
• TChart support from Steema Software.
• Optionally integrates with Web Broker and Web Snap
• Over 50 visual components.
• Many third party IntraWeb components available.
• Easy deployment, easy debugging!
• Not auto state management, but transparent!

**From Design Time to Browser in Seconds**

Here is the famous fishfact at design time.

**Design Time**

And here it is seconds later in the Browser

**Run Time**
You can click on the grid to sort, or select. It's even editable, click the edit button and the fields change to edits.
Browsers
IntraWeb supports Internet Explorer, Netscape 4, and Netscape 6. IntraWeb will also support Mozilla when released. IntraWeb also has extensive Netscape 4 support, which many tools ignore. Netscape 4, like it or not still accounts for 5% to 10% of the public Internet. More information on IntraWeb’s browser support can be found on the “Compatibility” page on the website.

How Flexible is IntraWeb?
Do not let IntraWeb’s ease of use fool you into believing it is not flexible. Many people have made this mistake, just as many people made this mistake about Delphi when it first appeared.

Delphi 1 was incredibly easy to use and extremely powerful. Many people who looked at Delphi assumed that because it was so easy to use, that it was not flexible or powerful. Delphi wrapped the Win API with the VCL. Visual C++ programmers and others scoffed at this fact and said, “I need the Win API. I will not be locked into some proprietary system.”

Just think if you had made that mistake and stayed with your previous tool. Yes, Delphi isolates the developer from the Win API with the VCL. The VCL makes developing
much easier than making all the Win API calls. Does using the VCL prevent you from using the Win API? Certainly not. The VCL just makes 98% of your tasks easier and eliminates the need for you to deal with the intricacies of the Win API. But anytime the VCL falls short you are free to call the Win API yourself to extend the VCL.

IntraWeb is the same. IntraWeb is merely a component architecture that interfaces to the HTTP and HTML layer for you. Think of it as VCL for web development. Just like the VCL, it does not lock you into it, you can easily interact at the HTML layer any time you choose.

**Custom Components**

IntraWeb also has an easy to use custom component layer. Writing custom components for IntraWeb is nearly as easy as writing normal Delphi components. Simply descend from TIWControl or an existing IntraWeb control, override a few methods and you are done! The IntraWeb core system does most of the work for you by managing state, life cycle, placement, and all communication details.

**Application Mode and Page Mode**

IntraWeb supports two modes for development, Application Mode and Page Mode. In short, Application Mode is easier to use at the expense of some flexibility. Page Mode is more flexible, but at the cost of ease of use. Consider the difference between programming in assembly and Delphi. Delphi is easier to use, but at the cost of some flexibility. Assembly is more flexible, but at the cost of ease of use. The difference between Application Mode and Page Mode is not as vast as the difference between Delphi and Assembly, but you get the picture.

More information on Application Mode and Page Mode can be found in “Modes” on the website.

**Layout and Positioning**

IntraWeb supports pluggable layout managers. Layout managers are used to position the controls on a page. IntraWeb supports three layout managers currently: Form, Template, and Flow. You can also add custom layout managers by descending from the base layout manager class.

Layout is specific to each form, and can be changed at any time by changing the layout manager.

**Form Layout**

By default when you create an IntraWeb form no layout manager is specified. When no layout manager is specified, IntraWeb creates an implicit Form Layout Manager which makes the web page look just like the form.
Template Layout

Template layout allows the presentation to be separated from the implementation. This allows you to develop, and your web designers to do the artistic design.

One template can be used for all browsers, or you can even specify specific templates for each browser type.

At design time, you create the controls that you need to change dynamically, or receive input from.

Design Time
Next an HTML template is created by your web designer. The template is shown below in Internet Explorer for screen capture purposes only. It was created in DreamWeaver, but could as easily have been done in Front Page, Namo, Notepad, edln, or any editor of your choice.

Template
At runtime the template processor automatically merges the template and the form automatically. You still have full control over the controls on your form including visibility. You can also embed extra tags in the template and handle them in the OnUnknownTag event.
Flow Layout
Flow layout is merely another layout manager. Flow layout is a feature that will be part of IntraWeb 5.0.

Flow layout is very similar to template layout except that the templates are kept internally to the application, and it includes a WYSIWYG HTML editor.

Here is the same form as in the template example. The only difference is that we have changed the layout manager to a flow layout manager.
Now if we double click on the layout manager component it launches the WYSIWYG HTML editor. Now you might say “Wait a minute! Its not in Delphi.” Yes, it is. The editor works with Delphi. You can add components, remove components, use the property inspector, etc. It is merely an alternate view of the form. This is a feature, not a design deficiency. This was purposefully designed this way to keep the layout management pluggable and changeable.

WYSIWYG HTML Editor
(Early Beta Screenshot, subject to change)
In Browser

Transparent State Management
Many web tools contain automatic state management. However, with automatic state management you still must deal with state. IntraWeb has transparent state management.

How do you deal with state in a normal Delphi application? You don’t you say? Exactly. That is how you deal with it in IntraWeb too.

Deploy Stand Alone
IntraWeb applications can be deployed as ISAPI DLLs, Apache DSOs, and stand alone executables. The stand alone versions are ideal for debugging, but they can be deployed as well. You can deploy them as a standard executable, or install them as a service. The stand alone versions are completely self sufficient and do not require any web server to be installed.

You can also convert between any of the deployment types easily by changing two lines of code.

More questions?
Please join our newsgroups where you will receive a prompt answer.