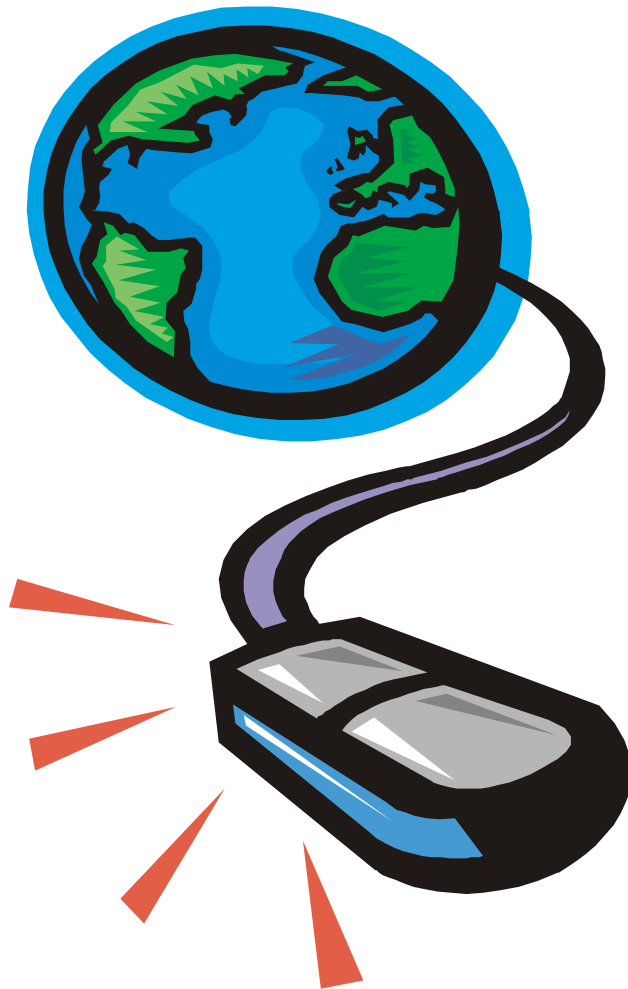


kbmWABD

ISAPI Web Server

Getting Started Guide



Kim Madsen
kbm@optical.dk
<http://delphi.e-indexit.com/>

Revised August 20, 2001

Installation

Compatibility

kbmWABD has been compiled and running successfully under Delphi 3/4/5/6 and Borland C++ Builder 3/4.

Files

- 1) Unzip kbmWABD_xxx.zip in an empty directory (eg. projects\kbmWABD) where xxx is the version number.
- 2) Get the resource_xxx.zip file from the place you downloaded kbmWABD_xxx.zip and unzip it in a subdirectory named resources (eg. projects\kbmWABD\resources)
- 3) Get the WWWserver_src_xxx.zip file from the place you downloaded kbmWABD_xxx.zip and unzip it in a subdirectory named WWWServer (eg. projects\kbmWABD\WWWServer)
- 4) Get the basic_xxx.zip file from the place you downloaded kbmWABD_xxx.zip and unzip it in a subdirectory named basic (eg. projects\kbmWABD\basic)

WWWserver_src_xxx.zip

Contains an ISAPI capable webserver which can be used for easy debugging ISAPI applications. The webserver can also be downloaded as an exe file. Check <http://delphi.e-indexit.com>

resources_xxx.zip

Contains some gifs and javascript files for support of the TWABD_MenuTree component.

kbmWABD_xxx.zip

Contains the complete source code for WABD in the kbm incarnation (2.xx and newer).

basic_xxx.zip

Contains a skeleton D5 project for a simple Hello world ISAPI application. This is the same application that is built in the tutorial.



Package Installation

Delphi 6

- 1) Open the runtime package kbmWABDD6Run.dpk and compile it.
- 2) Open the design time package kbmWABDD6Des.dpk and compile and press the Install button.

Delphi 5

- 1) Open the runtime package kbmWABDD5Run.dpk and compile it.
- 2) Open the design time package kbmWABDD5Des.dpk and compile and press the Install button.

Delphi 4

- 1) Open the kbmWABDD4.dpk file, compile it and press the Install button.

Delphi 3

- 1) Open the kbmWABDD3.dpk file, compile it and press the Install button.

C++Builder 4

- 1) Open the kbmWABDC4.bpk file, compile it and press the Install button.



kbmWABD Components

After installing the package a new kbmWABD tab will appear in the component palette.



Important: All kbmWABD components must be placed on the **TDatamodule** and not on a TForm.

The kbmWABD Components:



TWABD_SessionMgr

Handles all request from the ISAPI server and directs the requests to the right session or creates a new as needed. Further it handles statistics and a lot more. There must be only one session manager component and it must be placed on a TDatamodule by itself (possibly together with a TWABD_Admin component). A TDatamodule with a TWABD_SessionMgr on it is called a session manager module.



TWABD_Session

Handles a specific client session. A number of TWABD_Session's will be automatically created by the session manager (TWABD_SessionMgr). TWABD_Session must not be placed on a session manager module. Only one TWABD_Session can be on a TDatamodule. There can be (but usually are not) several TDatamodule's with each one TWABD_Session. The sessionmanager can then decide which session datamodule should be create for different applications within the same ISAPI dll. (The administration module is implemented that way). A TDatamodule with a TWABD_Session on it is called a session module.



TWABD_Admin

Is used to set up diverse administrative settings. If its used, it must be placed on a session manager module.



TWABD_Setup

Is used to specify paths for images and files. It must be placed on a session module and there can be more than one TWABD_Setup on it.



TWABD_Frameset

Is used together with TWABD_Frame to specify frame layouts. It must be placed on a session module and there can be more than one TWABD_Frameset on it.



TWABD_Frame

Is used together with TWABD_Frameset to specify frame layouts. It must be placed on a session module and there can be more than one TWABD_Frame on it.





TWABD_MenuTree

Is used to generate a javascript based menutree on the client. It must be placed on a session module.



TWABD_Form

Is used to generate advanced forms using a form editor (dbl click to open the editor). It must be placed on a session module. There can and often will be more than one TWABD_Form on the same session module. When the editor is open, a host of other socalled toplevel objects are available. When a TWABD_FormSection is added to the TWABD_Form, a list of section level objects becomes available.



TWABD_Javascript

Is used to contain Javascript code. It is the referenced from a TWABD_Frameset, a TWABD_Frame or a TWABD_Form. There can and often will be more than one TWABD_Javascript on the same session module.



TWABD_HTML

Is used for holding raw HTML code. The HTML code can contain variables of the form [!--VariableName--!] which will be automatically changed by the session manager module to the contents of the variable if found. Variables are setup using the property editor or at runtime by `ASessionMgr.VariableByName['VariableName']:= 'Some value';`



TWABD_HTMLFile

Is used for referencing an external HTML file. The file contents can be cached if needed.



TWABD_ExternalFrame

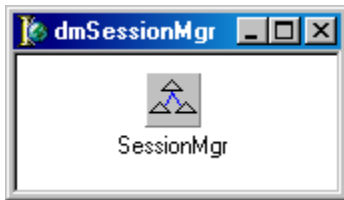
Is used for directing output to a frame external to kbmWABD or a new browser window.



Tutorial: Building a simple ISAPI application

To create a “Hello World” ISAPI application, do the following:

- 1) Create a new **DLL** project.
- 2) Create a new **TDatamodule** and name it **dmSessionMgr**.
- 3) Create a new **TDatamodule** and name it **dmSession**.
- 4) Put a **TWABD_SessionMgr** on the dmSessionMgr module. Name it **SessionMgr**.



- 5) Put a **TWABD_Session** on the dmSession module. Name it **Session**.
- 6) Add the **Session** unit to the SessionMgr data module.
- 7) Add code to the event handler for **OnCreateSession** on SessionMgr:

```
procedure TdmSessionMgr.SessionMgrCreateSession(  
    var NewSession: TWABD_Session; Request: TWABD_CustomRequest);  
begin  
    with TdmSession.Create(nil) do  
        NewSession := Session;  
end;
```

- 7) Add code to the event handler for **OnDestroySession** on SessionMgr:

```
procedure TdmSessionMgr.SessionMgrDestroySession(Session: TWABD_Session);  
begin  
    Session.Owner.Free;  
end;
```

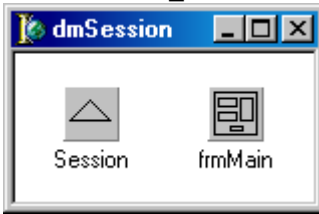
- 8) Add code to the end of the dmSessionMgr unit, just before the end. statement:

```
initialization  
    dmSessionMgr := TdmSessionMgr.Create(nil);
```

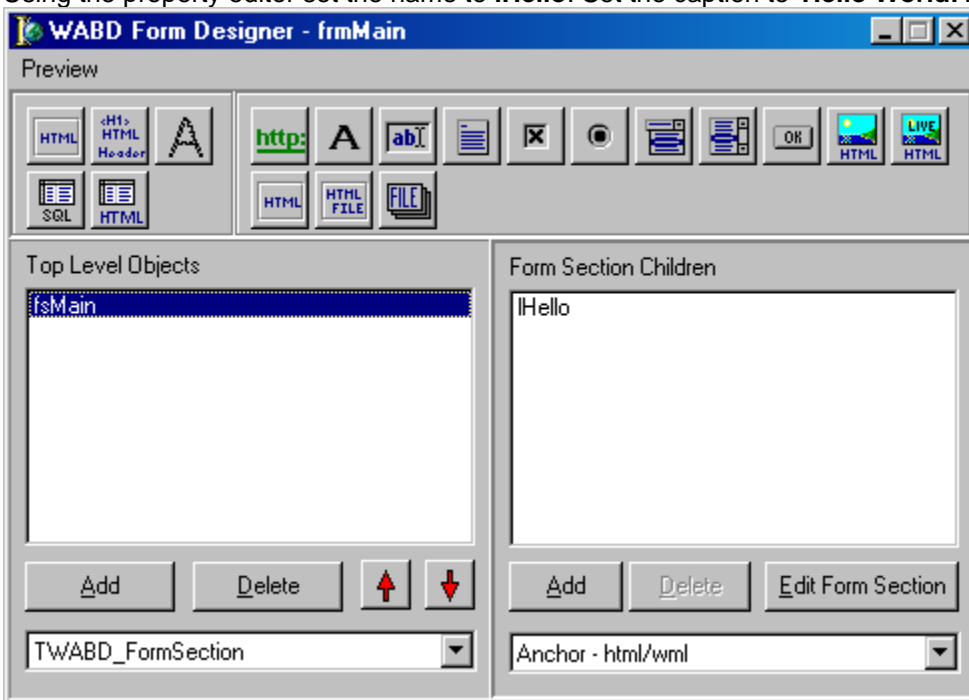
- 9) Add **WABD_Request** to the dmSessionMgr unit's uses clause.



- 10) Put a TWABD_Form on the dmSession module. Name it **frmMain**.



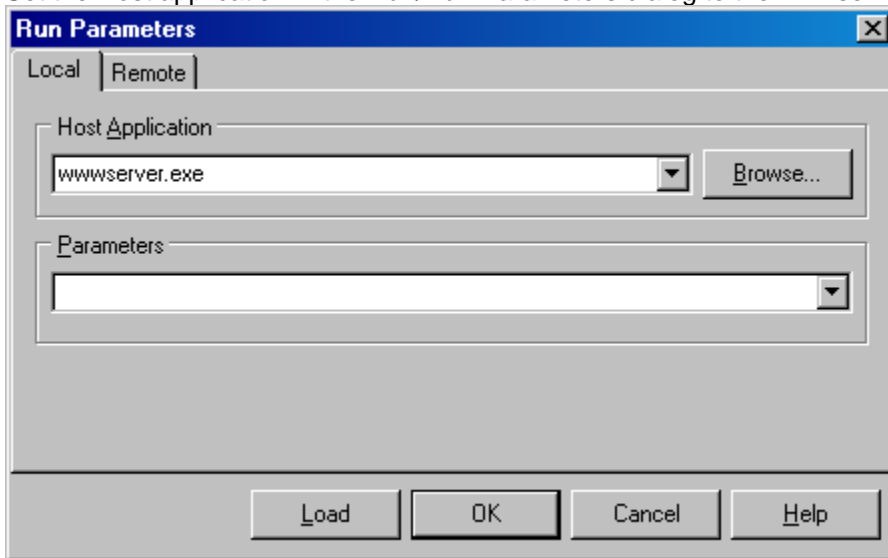
- 11) Double click frmMain (or right click it and choose 'Design WABD form').
- 12) Add a TWABD_Formsection toplevel object and select it in the list. Press F11 to open the property editor, and set the name to fsMain.
- 13) Now add a TWABD_Label section object (child object) to the framesection fsMain. Using the property editor set the name to **lHello**. Set the caption to **'Hello World!'**.



- 14) Set the **Session** component's **MainBody** property to **frmMain**.
- 15) Make sure the WABD install directory that contains the WABD dcu/pas files, are in the Library Search Path.
- 16) Save the project using any name you like. In this example we called it "HelloWorld.Dpr". This is the name of the dll that is loaded in the web browser.




- 17) Set the Host application in the Run/Run Parameters dialog to the wwwserver.exe.



- 18) Build the project but don't run it just yet.
- 19) If you have not compiled the wwwserver.dpr, open the wwwserver.dpr file in the WWWserver subdirectory and compile it. This wwwserver.exe must be on your path or copied to the same directory as your HelloWorld.Dll.

Now reload the HelloWorld.Dpr project you had just saved.

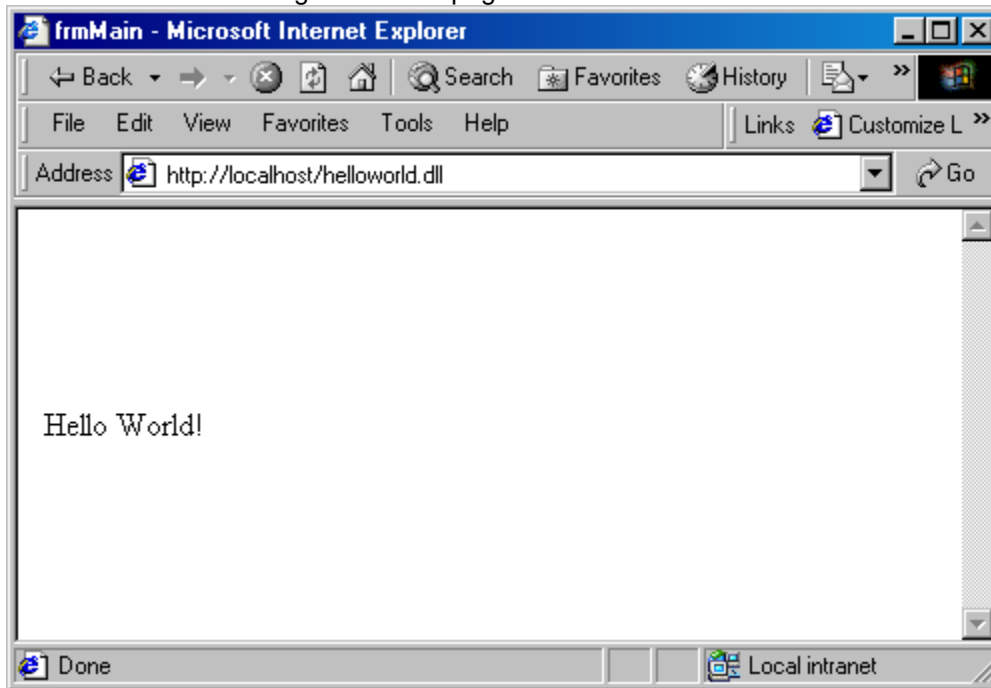
- 20) Press  to run the HelloWorld dll project that is currently loaded in the Delphi IDE. Don't worry that it is a dll. In a few seconds a test web server *wwwserver.exe* will start in the toolbar of the task manager. It's the light blue "WWW" icon shown below.



Later on you can right click on this blue webserver icon to shut it down or to open a administration window to see what's running. But for now we can ignore it.



- 21) Start your web browser (Netscape or MS Explorer) and enter the address: **localhost/helloworld.dll**. The webserver will then generate the page defined in the helloworld.dll.



Congratulations. You've just completed your first ISAPI web server. Not bad for 15 minutes of work.

The position of "Hello World", its font size and other properties can be set using the Object Inspector as shown on the following page.

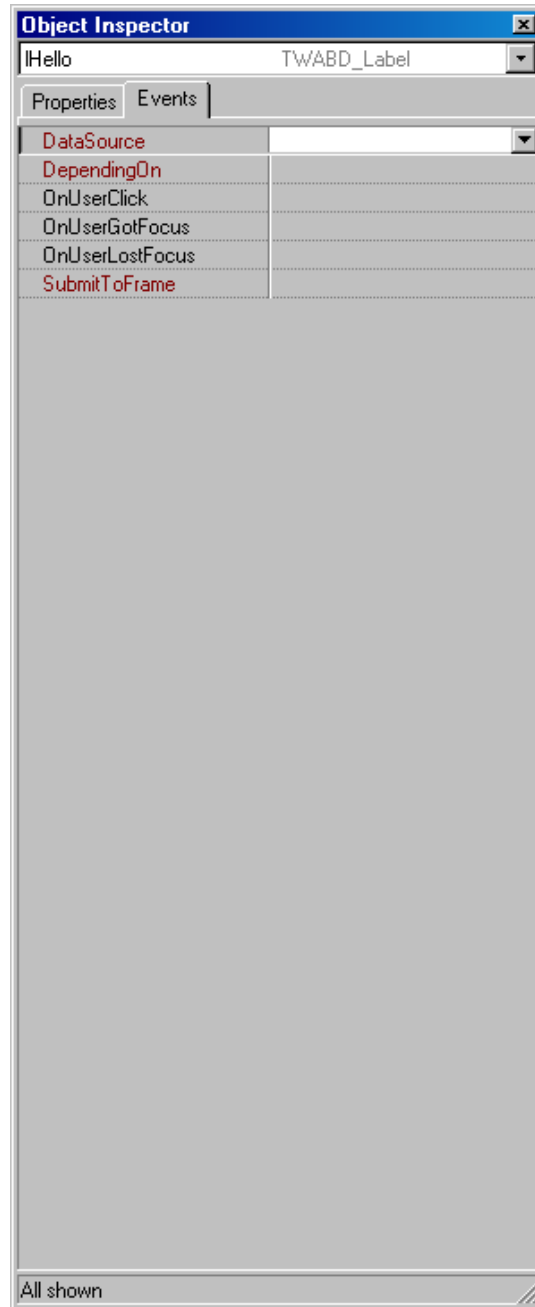
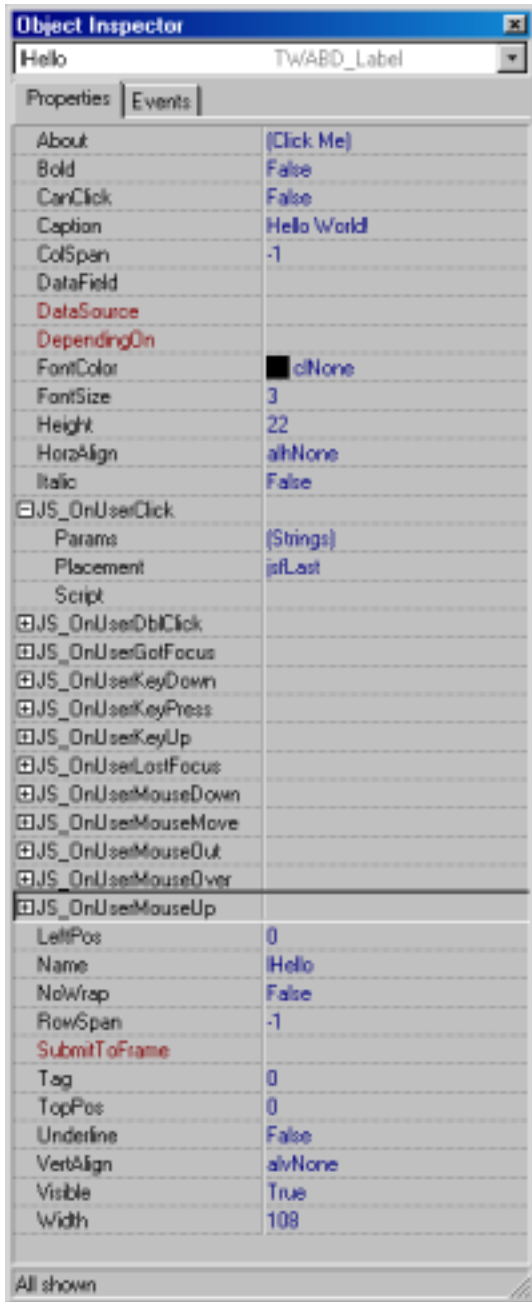
Shutting Down

If you want to stop this program and return to Delphi, right click on the blue webserver icon in the task bar and select **Shutdown**.



More Power with kbdWABD as Delphi/CB Web Servers

The real power of kbmWABD will be realized once you start to unlock the java scripts, properties and events that are available to you in the Object Inspector. This one little “Hello Word!” label has all of these java scripts, properties and events available:



All of these properties and events are available to the “Hello World” label.



By having a web server written using kbmWABD, you have all the power of Delphi or CB at your disposal. Why write web servers any other way?.

Feel free to experiment with the other kbmWABD components.

Please remember that these components can only be used in a TDatamodule and not on a TForm.

More Demos

If you have not already done so, please download the sample projects from <http://delphi.e-indexit.com> . These demos will show you what kbmWABD can do.

More Tutorials

If you would like to add your own tutorial to this manual, please contact Kim Madsen at the e-mail address shown on the cover page.

We need a screen shots of the web page(s) as seen from the web browser, along relevant screen shots of the TDatamodule's. We will also need a set of numbered instructions on how it was achieved using kbmWABD. The layout should be similar to the tutorial in this Getting Started Guide. Don't worry about the text formatting, spelling or grammar, we'll handle all that for you.

If you have a site up and running with kbmWABD, please let us know so we can include it on our website and documentation.

In the mean time, have fun with kbmWABD and show us what you can do. 😊

