

Tips for creating applications for Neuron.

Appearance

There are a few things you can do to make sure that your Neuron application appears to be a seamless part of your web page.

- Set the caption bar and border of the main window to none in the Viewer Properties dialog.
- Check the Center Client option in the main window properties.
- Set the default state of the main window to maximized.
- For a truly seamless appearance, set all of the following to the same color (such as gray).
 - The fillColor of the application's backgrounds.
 - The matColor of the main window (the mat is the area of the window interior not covered by the page if the window is larger than the page).
 - The background color of your web page.

Set them to one of the standard Windows colors, otherwise it may not look correct on 256-color systems.

- Be aware that different people will be using different video resolutions. Assume that most people will be running at 640x480, or at most 800x600 pixels. Set your page size with this in mind as well as the fact that part of the screen will be taken up with the browser's screen elements (toolbar, caption bar, menu, etc.).

Organization of Files

In order to ensure that your application works properly and can find all of its files, follow these guidelines.

- Set up the files on your hard drive in the same directory structure that you will be using on the web server.
- Before making a clip of any media file, copy that file to either the directory your book is in on the hard drive, or a subdirectory of that directory. This ensures that the book path will provide at least a partial path to the clip. Doing this will actually eliminate the need to run MediPackager.
- Put all of the books, system books, and/or DLL's that make up your application in the same directory.

Design Tips

Here are some guidelines for designing an application so that it runs efficiently and can download its data quickly.

- If your application is large, break it up into smaller books so that the entire application does not need to be downloaded all at once. For example, if your application has several "chapters," make each chapter a separate tbk file.
- Design your application keeping in mind the type of connection that most of your users will be using. If your application will be viewed over a local network using Ethernet, then bandwidth is not much of a consideration. Your application can contain large graphics and use large media files and still transfer the data acceptably fast. If most people will be connecting by modem, then you should refrain from using large graphics and large clip files. For example, use 16-color graphics since they are only one fourth the size of 256-color graphics. Use small sound clips for things like notification events (beeps, whistles, and the like).

- Pre-load clips that will be used several times as well as those that need to play immediately. Do this using the mmOpen command when your book first loads and do not use the mmClose command.
- When creating sound files, do not use full CD-quality (44.1 kHz, 16-bits per sample, stereo). Reducing the frequency rate, the sample size, or going from stereo to mono, will reduce the size of the file dramatically. For example, going from 44.1 kHz to 22 kHz, or going from stereo to mono, will cut the size of the file in half. Going from full CD-quality to 11 kHz, 8-bits per sample, and mono, will cut the size of the file to one sixteenth of its original size.

Coding Tips

Follow these coding guidelines to ensure that your application runs properly, runs efficiently, and can find all of its files.

- If using any system book or linking any DLL that comes with the ToolBook II runtime, use sysToolBookDirectory to provide the full path to the files on the user's hard drive. There is no need to put the files on the server since they will be installed in the user's runtime directory when Neuron is installed. For example:

```
linkDLL sysToolBookDirectory & "tb50win.dll"
push sysToolBookDirectory & "tb50hyp.sbk" onto sysbooks
```

- If using your own system books or DLL's, put them in the same directory on the server as your book and use just the file name in your code. Do not provide a full path of any kind as this will cause Neuron to look on the user's hard drive only and never check the web server. For example:

```
linkDLL "mydll.dll"
push "mysysbk.sbk" onto sysbooks
```

- If your application consists of multiple books, use only the file name when moving from book to book. For example:

```
go to page "chapter1" of book "chap1.tbk"
```

If you are using ToolBook II's built-in hyperlink capability, it should handle it for you automatically.

- Use only lower case letters in file names. This is because most web servers, especially those that run on Unix systems, are case-sensitive when it comes to file names and the capitalization must match exactly.

Deployment

Here are some tips for deploying your application on a web server.

- If possible, use a method, such as an FTP utility, that will copy the contents of an entire directory and its subdirectories. The FTP utility that comes with ToolBook II will do this for you. This will ensure that the directory structure on your hard drive is maintained on the server.
- When copying the files, use the option to copy all file names in lower case if it is available. Again, the FTP utility that comes with ToolBook II has this option. This ensures that the files will always be found since you will have used lower case file names in your code (for books, system books, and DLL's), and ToolBook stores clip file names in lower case (for media clips).

HTML Tips

Many Neuron applications will be embedded within a web page. Here are some guidelines on setting up your HTML document:

- Set the height and width settings in the <EMBED> and/or <OBJECT> tags to at least the same size as the largest page your application will display in the main window. Since pages are measured in page units, and the height and width of the Neuron display area is measured in pixels, you will need to convert from one to the other. On a system using a small fonts video driver, there are 15 page units per pixel. On a system using a large fonts video driver, there are 12 page units per pixel. To ensure that the size is large enough for everyone, assume that they will be using a large fonts driver. So, for example, if the page is 9600 page units wide and 7200 page units tall, then it will be 640x480 pixels with small fonts and 800x600 pixels with large fonts. You will want to take this into consideration when designing your application.
- If your application requires Non-Secure mode to operate correctly, inform the user of this on a preliminary page that has a link to the page with the application. This will give the user a chance to change to Non-Secure mode, or decide not to run the application altogether, before it downloads and runs.
- If your application runs correctly in Secure mode, use the ForceSecureMode option available in the ActiveX version of Neuron and the updated Netscape plugin. This will cause the application to run in secure mode regardless of what mode the user's system is set to. Also, if the user is set to Non-Secure with Confirmation, the confirmation dialog will not appear.
- Set up your HTML pages so they will work with both the ActiveX and Netscape plugin versions of Neuron. Here is an example:

```
<OBJECT ID="NeuronX1" WIDTH=640 HEIGHT=480
CLASSID="CLSID:F4818F4C-BEC5-11CF-83AD-00A0242FBEA6">
  <PARAMNAME="_Version" VALUE="65536">
  <PARAMNAME="_ExtentX" VALUE="22578">
  <PARAMNAME="_ExtentY" VALUE="16933">
  <PARAMNAME="_StockProps" VALUE="1">
  <PARAMNAME="LaunchBook"VALUE="kiosk.tbk">
  <EMBED SRC=kiosk.tbk WIDTH=640 HEIGHT=480>
</OBJECT>
```

Netscape will ignore the <OBJECT> and </OBJECT> tags. It will also ignore the <PARAM> tags since they are not between an <APPLET> and </APPLET> tag pair (which is used for Java applets). It will use the <EMBED> tag to launch the Neuron application. Internet Explorer will use the <OBJECT> and <PARAM> tags to run the Neuron application and will ignore the <EMBED> tag since it is between an <OBJECT> and </OBJECT> tag pair. The CLASSID setting is the Guaranteed Unique ID (GUID) for Neuron and will always be the same.

Server Configuration

The web server that is hosting your Neuron application must be configured to serve files with the extension TBK with the MIME type "application/toolbook." If it is your server system, consult the documentation for your server software. If the server belongs to someone else, such as an Internet Service Provider, you will need to ask them to configure the server for you. You can assure them that no additional software is required to run on their server. Only the MIME type needs to be configured which will have no impact on the server's operation.

The MIME type is necessary for Netscape so that it knows what plugin to use for the file. It is not necessary for Internet Explorer since the GUID tells it that it should use the ActiveX version of Neuron.

Hybrid Web/CD Applications

One of the more intriguing possibilities available with Neuron is the "hybrid" application where part of the application runs from the net and the rest resides on a CD-ROM that the user has in their local system. For example, you could put some or all of the media (sound files, music files, etc) on a CD and then access it from an application running in Neuron. The following script is designed for applications whose clips reside on a CD and should work just fine in applications designed for Neuron. It locates the CD in the user's CD-ROM drive and determines its drive letter. It then sets the CDMediaPath and HDMediaPath properties of the book so that it can find media on the CD.

```
to handle createCDMediaPath
  linkDLL sysToolBookDirectory & "tb50DOS.DLL"
  STRING getCDDriveList()
  STRING getFileOnlyList (STRING, STRING, STRING)
end

get setErrorMode(1)
allCDDrives = getCDDriveList()
cdDrive = checkCDDrive( allCDDrives)

step i from 1 to 3          -- 3 Tries.
  if cdDrive = -1          -- Error, file not found.
    request "The CD is required for this " \
    &"program. Place it in the CD-ROM " \
    &"drive now then press OK." \
    with "OK" or "Quit"
    if It = "Quit"
      send exit
      get setErrorMode(0)
      break to system
    end
    pause 5 seconds        -- Give CD-ROM time to \
                          -- read CD.
    cdDrive = checkCDDrive( allCDDrives)
    if i = 3 and cdDrive = -1
      request "Unable to locate CD, " \
      &"exiting."
      get setErrorMode(0)
      send exit
    end
  end
end
end
```

(continued)

```

get setErrorMode(0)
-- The following line should be edited to reflect the
-- directory structure on your CD-ROM.
if cdDrive <> -1 -- No error, file found.
    CDMediaPath of this book = cdDrive & \
        ":\instruct\samples;" \
        & cdDrive & ":\Video"
-- The following line is optional. This is to guarantee
-- that if a clip had HDMediaPath set instead of CDMediaPath
-- that it will be found on the CD ROM or the same path
-- structure that you defined for the CDMediaPath.
    HDMediaPath of this book = CDMediaPath of this book
    push "<Book Path>" onto HDMediaPath of this book
end
end

to get checkCDDrive allCDDrives
    step i from 1 to textLineCount( allCDDrives)
        currDrive = textLine i of allCDDrives

        -- The following lines check for a specific file on
        -- the CD-ROM. Change the filename and directory to
        -- reflect your application.
        get getFileOnlyList( currDrive \
            & ":\instruct\samples\*.tbk", "", "" )
        if It contains "cbt1" -- Change filename here.
            return currDrive
        end

    end
    return -1 -- Error, file not found.
end

to handle enterApplication
    linkDLL "Kernel"
        INT SetErrorMode(INT)
    end
    --Your other code can go here.
    --Add the following line to the enterApplication handler.
    send createCDMediaPath
    --Your other code can go here.
end

```

Check: <http://www.asymetrix.com> for latest news about Neuron
 Email to: techsup@asymetrix.com for technical assistance

Rev. 1.0 - 18 Sept. 96 by Nevada Hamaker