

ThinHelp demos and authoring ideas

```
{ewc THNHLP,THIN,!
  THDEMOS.LZH;CL
  OSBOOK.BMP;0/0/0
  /0/0/255/0;;JI(`~`stat
  icmenu')}} Static
embedded windows
(BMPs)
```

```
{ewc THNHLP,THIN,!
  THDEMOS.LZH;CL
  OSBOOK.BMP;0/0/0
  /0/0/255/0;;JI(`~`but
  tons')}} Embedded
window buttons
```

```
{ewc THNHLP,THIN,!
  THDEMOS.LZH;CL
  OSBOOK.BMP;0/0/0
  /0/0/255/0;;JI(`~`ani
  menu')}} Animated
embedded windows
```

```
{ewc THNHLP,THIN,!
  THDEMOS.LZH;CL
  OSBOOK.BMP;0/0/0
  /0/0/255/0;;JI(`~`thw
  ater')}} Watermarks
and background
colors
```

```
{ewc THNHLP,THIN,!
  THDEMOS.LZH;CL
  OSBOOK.BMP;0/0/0
  /0/0/255/0;;JI(`~`wa
  v')}} WAV audio
playback
```

```
{ewc THNHLP,THIN,!
  THDEMOS.LZH;CL
  OSBOOK.BMP;0/0/0
  /0/0/255/0;;JI(`~`ma
  cros')}} Custom
macros
```

```
{ewc THNHLP,THIN,!
  THDEMOS.LZH;CL
  OSBOOK.BMP;0/0/0
  /0/0/255/0;;JI(`~`win
  size')}} Window
```

This demo uses a series of shared bitmaps and animations to demonstrate a variety of authoring techniques which may be applied using ThinHelp's embedded window controls and custom macros. The **Copy** buttons will copy a template macro or embedded window statement to the clipboard for pasting into your RTF source documents. Please remember that in most cases all template text in square brackets [] *must* be edited and the square brackets removed to correspond with your own graphics and macros.

sizing control

{ewc THNHLP,THIN,!
THDEMOS.LZH;CL
OSBOOK.BMP;0/0/0
/0/0/255/0;;JI(`~`met
afile')}} Metafile
overlays

{ewc THNHLP,THIN,!
THDEMOS.LZH;CL
OSBOOK.BMP;0/0/0
/0/0/255/0;;JI(`~`pan
ing')}} Paning effects

{ewl THNHLP,THIN,!THDEMOS.LZH;NOTE.BMP;0/0/0/0/0/255/0} **All features demonstrated in this file (except HLPLHA file extraction which requires the HLPLHA.DLL extension) are available in both the 16 bit and 32 bit version of THNHLP.DLL. JPEG features are available in THNHLP.DLL only; all other features are available in THNBMP.DLL.**

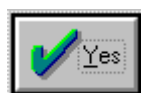
Static embedded windows

```
{ewc THNHLP,THIN,!
  THDEMOS.LZH;CLOSBOOK.BMP;0/0/0/0/0/255/0;;JI(`~`statica')}} Standard
embedded window .BMP bitmaps
```

```
{ewc THNHLP,THIN,!
  THDEMOS.LZH;CLOSBOOK.BMP;0/0/0/0/0/255/0;;JI(`~`staticshg')}}
Embedded window SHGs
```

```
{ewc THNHLP,THIN,!
  THDEMOS.LZH;CLOSBOOK.BMP;0/0/0/0/0/255/0;;JI(`~`jpstatic')}} Static
JPEG embedded windows (JPGs)
```

```
{ewc THNHLP,THIN,!
  THDEMOS.LZH;CLOSBOOK.BMP;0/0/0/0/0/255/0;;JI(`~`staticb')}}
Transparent bitmaps
```





```
{ewc THNHLP,THIN,!
THDEMOS.LZH;BUTTON_Y.SHG}
```

The example above includes the same visible hotspot as the first demo example, but you can't see the hotspot. ThinHelp does not support visible SHG hotspots, but the hotspots will still work.

The example below demonstrates the same SHG with its hotspot set to **invisible**, and it produces the identical effect to the first.

```
{ewc THNHLP,THIN,!
THDEMOS.LZH;BUTTON2Y.SHG}
```

{ewc THNHLP,THIN,!
THDEMOS.LZH;256THUMB
.SHG}

```
{ewc THNHLP,THIN,!THDEMOS.LZH;1.SHG}
```

As you will see by placing your cursor over the yellow hotspot, ThinHelp does not support hotspotted *metafile* SHGs. It will, however, support hotspotted *bitmap* SHGs.

Note also how this metafile-turned-SHG displays transparently, just as it does when displayed as a normal Help WMF metafile or metafile-based SHG.

**{ewr THNHLP,THIN,!
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th
demos.hlp>main'~`return')'~`CloseWindow(`'))}Watermarks with
hotspots (SHG version)**

This is one of the most interesting applications of ThinHelp's watermarking technology. This example shows an SHG version of the hotspotted watermark treatment. Paragraphs in this topic use a right indent of 1.1 inches (2,8cm) to allow for the colorful strip of hotspots which appears at right. It's somewhat tricky authoring, but we think you can appreciate how strikingly effective it can be. You should be able to mimic many of the effects of tabled or framed HTML pages with this technique.

The watermark is positioned using this macro to position the strip to the right and prevent it from scrolling:

**THWaterMark(hwndContext, 0, 0, coBackground, qchPath, `!
THDEMOS.LZH;NAVSTRIP', 0x1031)**

The flicker effect will be much more visible if you scroll using arrow keys rather than PageUp/PageDown.

It is critical that you set your margins correctly for this to work, although text overlap of hotspot regions is fine if you have no topic hotspots. Avoid overlapping hotspots wherever possible in your ThinHelp authoring.

{ewl THNHLP,THIN,!THDEMOS.LZH;TINOTRAN.BMP}Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. In human anatomy, the maxillary artery is the larger terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.

{ewr THNHLP,THIN,!THDEMOS.LZH;TINOTRAN.BMP}Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. In human anatomy, the maxillary artery is the larger terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.

{ewc THNHLP,THIN,!THDEMOS.LZH;TINOTRAN.BMP}Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. {ewc THNHLP,THIN,!THDEMOS.LZH;TINOTRAN.BMP} In human anatomy, the maxillary artery is the larger terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.{ewc THNHLP,THIN,!THDEMOS.LZH;TINOTRAN.BMP}

{ewc THNHLP,THIN,
THDEMOS.LZH;256THUMB
.BMP}

Left aligned:

{ewl THNHLP,THIN,!THDEMOS.LZH;CALLOUT.BMP}Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus.

Right aligned:

{ewr THNHLP,THIN,!THDEMOS.LZH;CALLOUT.BMP}Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus.

{ewc THNHLP,THIN,!
FLOWERJ.JPG}

Standard JPEG embedded window
(not added to a compressed archive)

```
{ewc THNHLP,THIN,!  
THDEMOS.LZH;FLOWERJ.JPG}
```

Standard JPEG embedded window
(from a compressed, baggaged
graphics archive)


```
{ewc THNHLP,THIN,+!  
THDEMOS.LZH;FLOWERJ.JPG}
```

Standard JPEG embedded window
cached to hard disk (performance
effect only observable on reboot.

.CAC file may be erased at any
time.)

```
      {ewc THNHLP,THIN,!
THDEMOS.LZH;FLOWERJ.JPG}{ewc
THNHLP,THIN,!THDEMOS.LZH;FLOW.PAL}
```

JPEG embedded window with 256 color palette control (.PAL file) (See previous examples for non-palette-controlled display results) This effect is achieved here using two separate embedded window statements, but both the .PAL and the .JPG can be referenced in the same statement if a partial filename is used as a filemask.

```
{ewr THNHLP,THIN,!
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th
demos.hlp>main'~`return')'~`CloseWindow(`'))}Transparency mask
demo
```

Original JPEG graphic:

```
{ewc THNHLP,THIN,!
THDEMOS.LZH;FLOWERJ.JPG}
```

2 color transparency mask:
(not normally referenced in the source)

```
{ewr THNHLP,THIN,!
THDEMOS.LZH;TMASK.BMP}
```

Result:

```
{ewc THNHLP,THIN,!THDEMOS.LZH;FLOWERJ}
combines mask with JPEG and adds a metafile
overlay if it is available for this bitmap.
```

```
{ewc THNHLP,THIN,!
THDEMOS.LZH;FLOWER;;;PI(`~`buttonp
op')}
```

```
{ewc THNHLP,THIN,!  
THDEMOS.LZH;REDROSE}
```

JPEG embedded window with
transparency mask. Achieved by
referencing REDROSE.JPG and
REDROSE.MSK simply as
REDROSE

```
{ewc THNHLP,THIN,!  
THDEMOS.LZH;FLOWER}
```

JPEG embedded window with
transparency mask and overlay.
Achieved by referencing
FLOWERJ.JPG, FLOWERJ.SEG
and FLOWER.MSK simply as
FLOWER

```
{ewc THNHLP,THIN,!  
THDEMOS.LZH;FLOWER}
```

JPEG embedded window with transparency mask, overlay and palette control file all referenced in the same statement. Achieved by referencing FLOWERJ.JPG, FLOWERJ.SEG, FLOWER.MSK and FLOW.PAL simply as FLOW

```
{ewc THNHLP,THIN,!
THDEMOS.LZH;THINICON.BMP;0/0/0/0
/0/128/128}
```

16 color transparent bitmap
(Transparency index entered as
0/128/128)

```
{ewc THNHLP,THIN,!
THDEMOS.LZH;THINICON.BMP}
```

Source bitmap without transparency

```
{ewc THNHLP,THIN,!
THDEMOS.LZH;SUNGLAS.BMP;0/0/0/0
/0/255/0}
```

256 color transparent bitmap
(Transparency index entered as 0/255/0)

```
{ewc THNHLP,THIN,!
THDEMOS.LZH;SUNGLAS.BMP}
```

Source bitmap without transparency


```
{ewc THNHLP,THIN,!
THDEMOS.LZH;FLOWERJ.JPG;5/40/0/0/255/0/0}
```

TrueColor transparent JPEG (not usually useful)
(Transparency index entered as red or 255/0/0 and the entire parameter is entered as 5/40/0/0/255/0/0.

Note: The color palette options specified in the display parameter had to be applied here because ThinHelp expects palette entry values for your JPEG if you enter any additional subparameters to control its display. The only way to avoid entering palette values in the first one or two parameters is to specify a .PAL file to be displayed with the JPEG.)

```
{ewc THNHLP,THIN,!
THDEMOS.LZH;SUNGL.BMP;0/0/0/0/
255/0}
```

Transparent 256 color bitmap with
added transparency mask. Achieved
(or, rather, *not* achieved) by referencing
SUNGLASS.BMP and SUNGL.MSK
simply as SUNGL

```
{ewc THNHLP,THIN,!
THDEMOS.LZH;GLASSES.BMP;0/0/0/0
/0/255/0}
```

Transparent 256 color bitmap with
metafile overlay. Achieved by
referencing GLASSES2.BMP and
GLASSES.SEG simply as GLASSES
This combination does not function
properly.

```
{ewc THNHLP,THIN,!
THDEMOS.LZH;GLASSES3.BMP;0/0/0/
0/0/255/0}
```

Transparent 256 color bitmap with
palette control (used in difficult situations
where palette distortion is hard to
control). Achieved (or, rather, *not*
achieved) by referencing
GLASSES3.BMP and GLASSES3.PAL
simply as GLASSES3 This
combination does not function properly.

```
{ewr THNHLP,THIN,!  
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th  
demos.hlp>main'~`return')'~`CloseWindow(`'))}Button demo
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;BFACE.BMP;0/0/2;;PI(`~`buttonpop')}
```

Standard 16 color
embedded window button

```
{ewr THNHLP,THIN,!  
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th  
demos.hlp>main'~`return')'~`CloseWindow(`'))}Button demo
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;BFACE256.BMP;0/0/2;;PI(`'~`buttonpop')}
```

Standard 256 color
embedded window button

Remember to add *two* semicolons after the display properties parameter prior to entering your macro, and to use tilde “~” characters in place of commas.

```
{ewr THNHLP,THIN,!  
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th  
demos.hlp>main'~`return')'~`CloseWindow(`')')})Button demo
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;REDROSE.JPG;5/40/2;;PI(`'~`buttonpop')}
```

TrueColor JPEG embedded window button. The display properties parameter is 5/40/2.

Note: The color palette options specified in the display parameter had to be applied here because ThinHelp expects palette entry values for your JPEG if you enter any additional subparameters to control its display. The only way to avoid entering palette values in the first one or two parameters is to specify a .PAL file to be displayed with the JPEG.)

**{ewr THNHLP,THIN,!
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th
demos.hlp>main'~`return')'~`CloseWindow(`'))}Button demo**

{ewc THNHLP,THIN,!THDEMOS.LZH;THINICON.BMP;0/0/2/0/0/128/128;;PI(`'~`buttonpop')}

Transparent embedded window button (not supported due to flicker effects when the button is
clicked on slower systems. Test thoroughly.)


```
{ewr THNHLP,THIN,!  
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th  
demos.hlp>main'~`return')'~`CloseWindow(`'))}Button demo
```

```
{ewc THNHLP,THIN,!STARTCAR.LZH;ST;0/0/2/150;;PI(`'~`buttonpop')}
```

Animated embedded window button

**{ewr THNHLP,THIN,!
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th
demos.hlp>main'~`return')'~`CloseWindow(`'))}Buttonized callout
box**

{ewc THNHLP,THIN,!THDEMOS.LZH;CALLOUT.BMP;0/0/2;;PI(`~`buttonpop')}

Callout box used as a large button. Can be used to offer extra information about a given topic.

```
{ewc THNHLP,THIN,  
STARTCAR.LZH;ST;0/0/0/150}
```

Basic animated embedded window.

Display properties: 0/0/0/150
(produces a 150ms frame advance
interval)

```
{ewc THNHLP,THIN,  
WORLD.LZH;WO;0/0/0/75/255/0/25  
5}
```

Correctly-displayed transparent
animation (all frames use the same
transparent regions)

Display properties:
0/0/0/75/255/0/255 (75ms per frame
advance rate, magenta specified as
transparent color)

{ewc THNHLP,THIN,!
BIRDTOON.LZH;BI;0/0/0/125/255/255/255}

Incorrectly-displayed transparent animation (frames
use differing transparent regions...must be displayed
nontransparent as shown below)

Display properties: 0/0/0/125/255/255/255 (125ms
frame advance interval, white specified as
transparent color)

{ewc THNHLP,THIN,!BIRDTOON.LZH;BI;0/0/0/125}

```
{ewr THNHLP,THIN,!  
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th  
demos.hlp>main'~`return')'~`CloseWindow(`'))}Animation demo
```

```
{ewc THNHLP,THIN,!CLOCK.LZH;CLOCK;0/0/25/125}
```

Animation with scroll bar; start when clicked or static and step through with scroll buttons (recommended method)

Display properties: 0/0/25/125 (125ms frame advance interval with play-once (1) + scroll bar (8) + start-when-clicked (16) = 25 as display type subparameter

{ewc THNHLP,THIN,!CLOCK.LZH;CLO;0/0/5/125}

Animation with metafile overlay on last frame.

Usually referenced as a play-once animation (4 for metafile overlay + 1 for single-play mode = 5 as the display properties subparameter)

Display properties: 0/0/5/125

```
      {ewc  
THNHLP,THIN,!THDEMOS.LZH;FLOWERJ.JPG/CALLOUT/BFA  
CE/GLASSES2.BMP/256THUMB.BMP;0/0/0/500}
```

User-defined animation sequence. Note specified order versus actual display order. Display order is determined by the order in which frames are added to the archive.

If you have a non-white text background specified, you'll notice white space during display of the smaller bitmaps. All bitmaps in user-defined sequences should be sized equally.

You will also notice that the "Click me" button face suffers from significant palette distortion due to the fact that the animation is deriving its palette from the first bitmap in the animation, and the rose's palette is incompatible with Click-me's palette.

Embedded window statement:

```
{ewc THNHLP, THIN, !THDEMOS.LZH;  
FLOWERJ.JPG/CALLOUT/BFACE/GLASSES2.BMP/256THUMB  
.BMP; 0/0/0/500}
```


{ewc THNHLP,THIN,
CLOCK.LZH;CLOCK;0/0/1/125}

Play-once animation. Reopen popup to replay.

Display properties: 0/0/1/125
(125ms frame advance interval, play once only)

**{ewr THNHLP,THIN,!
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th
demos.hlp>main'~`return')'~`CloseWindow(`'))}Animation demo**

{ewc THNHLP,THIN,!CLOCK.LZH;CLOCK;0/0/17/125}

Start-when-clicked animation (normally configured to play once).

Display properties: 0/0/17/125 (125ms frame advance interval, play-once (1) + start-when-clicked (16) = 17 as third subparameter)

**{ewr THNHLP,THIN,!
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th
demos.hlp>main'~`return')'~`CloseWindow(`'))}Animation demo**

{ewc THNHLP,THIN,!CLOCK.LZH;CLO;0/0/47/125;;PI(`'~`buttonpop')}

Everything but the kitchen sink (play once, transparent, buttonized, scroll bars, metafile on last frame)

Display properties: 0/0/47/125

This display type is usually defined as play-once (1) + buttonize (2) + metafile overlay on last frame (4) + scroll bar (8) + make nonprintable (32) for a total of 47 as the display type subparameter.

Multiple animations at differing speeds

```
{ewc THNHLP,THIN,!STARTCAR.LZH;ST;0/0/0/150}
```

Basic animated embedded window.

Display properties: 0/0/0/150

(produces a 150ms frame advance interval)

```
{ewc THNHLP,THIN,!
```

```
WORLD.LZH;WO;0/0/0/75/255/0/255}
```

Display properties: 0/0/0/75/255/0/255

(75ms per frame advance rate,

magenta specified as transparent color)

```
{ewc THNHLP,THIN,!
```

```
BIRDTOON.LZH;BI;0/0/0/400/255/255/255}
```

Display properties: 0/0/0/400/255/255/255

(125ms frame advance interval,

white specified as transparent color)

**{ewr THNHLP,THIN,!
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark('demo')~`JI('th
demos.hlp>main'~`return')'~`CloseWindow(''))}User-definable frame
advance rate**

```
{ewc THNHLP,THIN,!CLOCK.LZH;CLOCK}  
{ewc THNHLP,THIN,!THDEMOS.LZH;FAST.BMP;0/0/2/0;;THSetAnimationSpeed(75)}{ewc  
  THNHLP,THIN,!THDEMOS.LZH;SLOW.BMP;0/0/2/0;;THSetAnimationSpeed(250)}{ewc  
    THNHLP,THIN,!  
THDEMOS.LZH;STOP.BMP;0/0/2/0;;THSetAnimationSpeed(150000):THMB(hwndContext~`Anim  
  ation is now off. You may need to restart the helpfile in order to restart this particular  
    animation.'~`Animation stopped'~64)}
```

This effect only works when no frame advance interval, or an interval of “0” is specified in the embedded window statement. Also note the message attached to the Stop button, which can be authored as a popup or messagebox, which is presented as a courtesy to users. Higher values may be used for the Stop button's animation speed macro, but watch for erratic behavior from these buttons/hotspots.

Embedded window statements for the bitmap buttons (requires registration of the THSetAnimationSpeed() macro):

```
{ewc THNHLP,THIN,!THDEMOS.LZH;FAST.BMP;0/0/2/0;;THSetAnimationSpeed(75)}  
{ewc THNHLP,THIN,!THDEMOS.LZH;SLOW.BMP;0/0/2/0;;THSetAnimationSpeed(250)}  
{ewc THNHLP,THIN,!THDEMOS.LZH;STOP.BMP;0/0/2/0;;THSetAnimationSpeed(150000)}
```

{ewc THNHLP,THIN,!JPANI.LZH;TEXT}

Animated JPEG (uses 500ms frame advance interval specified in the jump macro in the previous topic. Take careful note of the display speed on first pass of the animation as the JPEGs decompress. This will be much slower on slower systems...longer frame advance intervals are recommended.)

Also note the lack of distortion in this sequence in 256 color mode even with multiple JPEGs of widely differing palettes. This is because ThinHelp has drawn the palette for the animation from the default Windows palette. So while the display is somewhat speckled for some of these textures, the distortion is no more severe than with a poorly palette matched Web page.

Watermarks and background colors

JPEG

{ewl THNHLP,THIN,!THDEMOS.LZH;NOTE.BMP;0/0/0/0/255/0}JPEG watermarks may be cached to hard disk by adding “+” in front of the LZH archive’s path. BMPs do not significantly benefit from caching and this parameter is recommended only for JPEGs.

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`thdemos.hlp>demo'~`wmsampa')}}
Wrap-to-window (Includes a .PAL file for color control defined as an embedded
window in this topic)
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`THWaterMark(hwndContext~0~
0~coBackground~qchPath~`[+! or PATH][ARCHIVE.LZH];
[FILENAME]'~0x200[n])):THMB(hwndContext~`Template text copied to clipboard.
Paste into your source document and edit to suit your project. Remember to
remove square brackets.'~`Template copied'~64)}}
THWaterMark(hwndContext,0,0,coBackground,qchPath,`[+! or PATH]
[ARCHIVE.LZH];[FILENAME]',0x200[n])
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`thdemos.hlp>demo'~`wmsampb')}}
Tiled JPEG, scrolls with topic text
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`THWaterMark(hwndContext~0~
0~coBackground~qchPath~`[+! or PATH][ARCHIVE.LZH];
[FILENAME]'~0x000[n])):THMB(hwndContext~`Template text copied to clipboard.
Paste into your source document and edit to suit your project. Remember to
remove square brackets.'~`Template copied'~64)}}
THWaterMark(hwndContext,0,0,coBackground,qchPath,`[+! or PATH]
[ARCHIVE.LZH];[FILENAME]',0x000[n])
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`thdemos.hlp>demo'~`wmsampc')}}
Tiled JPEG, does not scroll with topic text
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`THWaterMark(hwndContext~0~
0~coBackground~qchPath~`[+! or PATH][ARCHIVE.LZH];
[FILENAME]'~0x100[n])):THMB(hwndContext~`Template text copied to clipboard.
Paste into your source document and edit to suit your project. Remember to
remove square brackets.'~`Template copied'~64)}}
THWaterMark(hwndContext,0,0,coBackground,qchPath,`[+! or PATH]
[ARCHIVE.LZH];[FILENAME]',0x100[n])
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`thdemos.hlp>demo'~`wmsampd')}}
Tiled JPEG, does not scroll with topic text
```

Positioned JPEG, single instance, scrolling (uses .PAL file embedded in topic window for palette control)

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`THWaterMark(hwndContext~0~
0~coBackground~qchPath~`[+|! or PATH][ARCHIVE.LZH];[FILENAME]'~0x0[n][n]
[n])):THMB(hwndContext~`Template text copied to clipboard. Paste into your
source document and edit to suit your project. Remember to remove square
brackets.'~`Template copied'~64)}
THWaterMark(hwndContext,0,0,coBackground,qchPath, `[+|! or PATH]
[ARCHIVE.LZH];[FILENAME]',0x0[n][n][n])
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`thdemos.hlp>demo'~`wmsampd2')}}
Positioned JPEG, single instance, nonscrolling (produces significant flicker)
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`THWaterMark(hwndContext~0~
0~coBackground~qchPath~`[+|! or PATH][ARCHIVE.LZH];[FILENAME]'~0x1[n][n]
[n])):THMB(hwndContext~`Template text copied to clipboard. Paste into your
source document and edit to suit your project. Remember to remove square
brackets.'~`Template copied'~64)}
THWaterMark(hwndContext,0,0,coBackground,qchPath, `[+|! or PATH]
[ARCHIVE.LZH];[FILENAME]',0x1[n][n][n])
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`thdemos.hlp>demo'~`wmsampe')}}
Positioned watermark sampler (nine different positions)
```

Special effects

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`thdemos.hlp>demo'~`wmfxa')}}
Watermark slide show
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`THDEMOS.HLP>DEMO'~`wmfxb')}}
Conditional positioned bitmap display
```

```
{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;EP(`winhelp.exe -i
wmfxb2 thdemos.hlp'~0)}}User-selectable on/off watermark display
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`THDEMOS.HLP>DEMO'~`wmfxc')}}
Conditional on/off watermarks
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`THDEMOS.HLP>DEMO'~`wmfxc')}}
Watermarks and background colors
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`THDEMOS.HLP>DEMO'~`wmfxd')}}
Watermarks with hotspots
```



```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`thdemos.hlp>demo'~`shgdemd')}}
SHG graphic used as a hotspotted watermark in place of a .BMP + .SEG
combination
```

Background color control

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`THDEMOS.HLP>OTHRHELP'~`wmbg
a'))Background color sampler (indexed)
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`THDEMOS.HLP>OTHRHELP'~`wmbg
b'))Background color sampler (system coordinated)
```

```
{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;EP(`winhelp.exe -i
wmfxb2 thdemos.hlp'~0))User-selectable on/off background color display
```

```
{ewr THNHLP,THIN,!
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th
demos.hlp>main'~`return')'~`CloseWindow(`'))}Watermark test
{ewc THNHLP,THIN,!ANNEX.LZH;PAL}
```

Welcome to the annex

{ewr THNHLP,THIN,!}

THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark('demo')~`JI(`th
demos.hlp>main'~`return')'~`CloseWindow(''))}Watermark test

Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. In human anatomy, the maxillary artery is the larger terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.

Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. In human anatomy, the maxillary artery is the larger terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.

Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. In human anatomy, the maxillary artery is the larger terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.

Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. In human anatomy, the maxillary artery is the larger terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.

Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. In human anatomy, the maxillary artery is the larger terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.

Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. In human anatomy, the maxillary artery is the larger

terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.

```
{ewr THNHLP,THIN,!
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th
demos.hlp>main'~`return')'~`CloseWindow(`'))}Watermark test
```

{ewr THNHLP,THIN,!THDEMOS.LZH;BACK.BMP;0/0/2/0;;Exit()}}User-configurable watermark test

This sample topic, which uses a new instance of the demos helpfile for simplicity's sake, uses a button on the button bar to let the user decide whether or not to display the watermark. The idea is to have the button display the texture on demand, so clicking once turns the background texture on while clicking a second time turns it off. This complicated macro is usually added to the [CONFIG] section of the project file. It can be used to allow either user-selectable forced background color *or* background texture. You can actually allow both if you use a little imagination, but the complexity of allowing *three* variables is beyond the scope of this demo. This macro string is too long to be attached to a **Copy** button; either copy it whole from the window (Win95) or use the **Copy** function (Win 3.1/3.11/NT) to copy it to the clipboard.

Button-selectable background color

```
CreateButton('btn_clr','C&olor',`IfThenElse(IsMark('clr'),
`THWaterMark(hwndContext,0,0,[your color index or THRGB() macro],
qchPath,`,0x0000):DeleteMark('clr'),'`THWaterMark(hwndContext,
0,0,coBackground,qchPath,`IWM.LZH;WM.BMP',0x000[n]):SaveMark('clr'))')
```

Button-selectable background texture

```
CreateButton('btn_tex','&Texture',`IfThenElse(IsMark('tex'),
`THWaterMark(hwndContext,0,0,coBackground,qchPath,
`,0x0000):DeleteMark('tex'),'`THWaterMark(hwndContext,0,0,
coBackground,qchPath,`[+|! or PATH][ARCHIVE.LZH];[FILENAME]',0x[n][n][n]
[n]):SaveMark('tex'))')
```

{ewr THNHLP,THIN,!

**THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark('demo')~`JI('th
demos.hlp>main'~`return')'~`CloseWindow(''))}Watermark test**

{ewc THNHLP,THIN,!THDEMOS.LZH;FLOW.PAL}Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. In human anatomy, the maxillary artery is the larger terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.

Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. In human anatomy, the maxillary artery is the larger terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.

Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. In human anatomy, the maxillary artery is the larger terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.

Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. In human anatomy, the maxillary artery is the larger terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.

Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. In human anatomy, the maxillary artery is the larger terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.

Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. In human anatomy, the maxillary artery is the larger

terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.

{ewr THNHLP,THIN,!}

**THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark('demo')~`JI('th
demos.hlp>main'~`return')'~`CloseWindow(''))}Watermark test**

{ewc THNHLP,THIN,!THDEMOS.LZH;FLOW.PAL}Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. In human anatomy, the maxillary artery is the larger terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.

Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. In human anatomy, the maxillary artery is the larger terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.

Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. In human anatomy, the maxillary artery is the larger terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.

Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. In human anatomy, the maxillary artery is the larger terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.

Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. In human anatomy, the maxillary artery is the larger terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.

Amitosis is the anatomical term given to cell reproduction by direct division. The nucleus becomes constricted in the middle, forming an hour-glass shape and then divides into two. This is followed by a division of the whole protoplasmic mass of the cell; two daughter cells are thus formed, each containing a nucleus. In human anatomy, the maxillary artery is the larger

terminal branch of the external carotid artery. It rises behind the neck of the mandible and is at first imbedded in the parotid gland. It passes forwards between the neck of the mandible and the sphenomandibular ligament and then to the Lateral pterygoid and enters the pterygopalatine fossa between the two heads of that muscle.

```
{ewr THNHLP,THIN,!
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th
demos.hlp>main'~`return')'~`CloseWindow(`'))}Watermark position
sampler
```

Click the hotspots to set demo watermark positions and copy template text to clipboard.

	Left	Right	Center
Top	<u><u>0x001[n]</u></u>	<u><u>0x002[n]</u></u>	<u><u>0x003[n]</u></u>
			↓
Middle	<u><u>0x011[n]</u></u>	<u><u>0x012[n]</u></u>	<u><u>0x013[n]</u></u>
			↓
Bottom	<u><u>0x021[n]</u></u>	<u><u>0x022[n]</u></u>	<u><u>0x023[n]</u></u>
			↓

```
{ewr THNHLP,THIN,!
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th
demos.hlp>main'~`return')'~`CloseWindow(`'))}Watermark slide
show
```

Show popup again

[Show slide 1](#)

[Show slide 2](#)

[Show slide 3](#)

[Show slide 4](#)

[Show slide 5](#)

[Show slide 6](#)

[Show slide 7](#)

[Show slide 8](#)

[Show slide 9](#)

Here's a sample slide show that steps through the frames of the "Time's Up" animation. Clicking each of the "Show slide" buttons will display a different slide aligned to the left edge and vertically centered in this window. In order to heighten the effect, we've "darkened the room" by creating a forced black background using `THWaterMark(hwndContext,475,375,0x000000,qchPath,`,0x000000)`. This is an effect you can only get in 16 bit helpfiles at this time.

The base macro for this demo is `THWaterMark(hwndContext,0,0,0x000000,qchPath,`!CLOCK.LZH;CLOCKnn',0x0111)`. (We have to apply color control using **0x000000** with each slide because the normal **coBackground** parameter would otherwise knock out the black background.)

We also applied color control to the hotspot anchor text using "*" as the first hidden character so our links would use our chosen colors and also preserve the telltale hotspot underlining.

**{ewr THNHLP,THIN,!
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th
demos.hlp>main'~`return')'~`CloseWindow(`'))}Conditional
positioned bitmap display**

{ewc THNHLP,THIN,!THDEMOS.LZH;FLOW.PAL}This topic demonstrates ThinHelp's ability to display conditional bitmaps in topics using the THWaterMark() macro. If you browsed the watermark slide show before accessing this topic, a rose bitmap will be shown in the lower left corner of the window. If you have *not* browsed the watermark slide show topic, you'll see no watermark bitmap at all.

We achieved this effect by setting a mark in the slide show topic. If the mark is detected on entry to this topic, then the watermark is shown. If it is not detected, no watermark appears. Here's how the macro looks:

```
IfThen(IsMark(`condition'), `THWaterMark(hwndContext, 0,0,coBackground,qchPath, `!  
THDEMOS.LZH',0x0211)')
```

No template is supplied here because the applications of this macro are generally highly individualized to the project.

[Click here](#) to delete the **condition** mark if you'd like to see the effect both ways.

```
{ewr THNHLP,THIN,!
```

```
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th  
demos.hlp>main'~`return')'~`CloseWindow(`'))}Conditional on/off  
watermarks
```

This effect allows you to give the user control over the look of your Help. You can offer this as a popup or, if you have the tools and skills, use INI entries to store user-selected values. In this example we're letting you select a watermark, just like the slide show demo, but in this case it's tiled. The most courteous way to present this option to your user will be to include a choice of one or more plain background color tiles along with any graphical watermark(s). Many users may not prefer the distraction of even a subtle texture while they're trying to read.

Select a watermark:

ThinHelp logo

```
THWaterMark(hwndContext, 0,0,coBackground,qchPath, `!WM.LZH;WM.BMP',0x0000)
```

Plain white

```
THWaterMark(hwndContext, 0,0,0xFFFFFFFF,qchPath,``,0x0000)
```

Ergonomic green

```
THWaterMark(hwndContext, 0,0,0xC0FFC0,qchPath,``,0x0000)
```

{ewr THNHLP,THIN,!}

**THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark('demo')~`JI('th
demos.hlp>main'~`return')'~`CloseWindow(''))}Watermarks with
hotspots**

This is one of the most interesting applications of ThinHelp's watermarking technology. Paragraphs in this topic use a right indent of 1.1 inches (2,8cm) to allow for the colorful strip of hotspots which appears at right. It's somewhat tricky authoring, but we think you can appreciate how strikingly effective it can be. You should be able to mimic many of the effects of tabled or framed HTML pages with this technique.

The watermark is positioned using this macro to position the strip to the right and prevent it from scrolling:

**THWaterMark(hwndContext, 0, 0, coBackground, qchPath, `!
THDEMOS.LZH;NAVSTRIP', 0x1031)**

The flicker effect will be much more visible if you scroll using arrow keys rather than PageUp/PageDown.

It is critical that you set your margins correctly for this to work, although text overlap of hotspot regions is fine if you have no topic hotspots. Avoid overlapping hotspots wherever possible in your ThinHelp authoring.

Up you go...

Backing up...

Out of here...

Here you go.

There you are.

Bye, now.

{ewr THNHLP,THIN,!

THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th
demos.hlp>main'~`return')'~`CloseWindow(`'))}Background color
sampler (indexed)

This table represents a short list of recommended standard background colors for use with the background color control feature of THWaterMark(). Hotspots are live and will activate the selected color in this window and copy a template macro to the clipboard. Use the **Default** hotspot below to restore normal color if all screen text should become invisible.

Default

<u>Blue</u>	<u>Lime</u>	<u>Red</u>
<u>Aqua</u>	<u>Fuchsia</u>	<u>Yellow</u>
<u>Maroon</u>	<u>Navy</u>	<u>Teal</u>
<u>Green</u>	<u>Olive</u>	<u>Purple</u>
<u>Silver</u>	<u>Gray</u>	<u>Black</u>
<u>White</u>	<u>Orange</u>	<u>Blue-white</u>
<u>Green- white</u>	<u>Yellow- white</u>	<u>Aqua- white</u>
<u>Windows off-white</u>	<u>Ergonomi c paper 1</u>	<u>Ergonomi c paper 2</u>

{ewr THNHLP,THIN,!

THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark('demo')~`JI('th
demos.hlp>main'~`return')'~`CloseWindow(''))}Windows desktop
color coordinator

These background color values must be used with extreme care when applying specific colors (e.g. black) to your topic text. Hotspots are live and will activate the selected color in this window and copy a template macro to the clipboard. Use the **Default** hotspot below to restore normal color if all screen text should become invisible.

Default

<u>Scroll bar</u> <u>(0x100000</u> <u>00)</u>	<u>Desktop</u> <u>backgrou</u> <u>nd</u> <u>(0x100000</u> <u>01)</u>	<u>Active</u> <u>window</u> <u>caption</u> <u>(0x100000</u> <u>02)</u>
<u>Inactive</u> <u>window</u> <u>caption</u> <u>(0x100000</u> <u>03)</u>	<u>Menu bar</u> <u>backgrou</u> <u>nd</u> <u>(0x100000</u> <u>04)</u>	<u>Window</u> <u>backgrou</u> <u>nd</u> <u>(0x100000</u> <u>05)</u>
<u>Window</u> <u>frame</u> <u>(0x100000</u> <u>06)</u>	<u>Menu text</u> <u>(0x100000</u> <u>07)</u>	<u>Window</u> <u>text</u> <u>(0x100000</u> <u>08)</u>
<u>Caption</u> <u>text</u> <u>(0x100000</u> <u>09)</u>	<u>Active</u> <u>window</u> <u>border</u> <u>(0x100000</u> <u>a)</u>	<u>Inactive</u> <u>window</u> <u>border</u> <u>(0x100000</u> <u>0b)</u>
<u>Applicatio</u> <u>n</u> <u>workspac</u> <u>e</u> <u>(0x100000</u> <u>0c)</u>	<u>Highlighte</u> <u>d region</u> <u>(0x100000</u> <u>0d)</u>	<u>Highlighte</u> <u>d text</u> <u>(0x100000</u> <u>0e)</u>
<u>Button</u> <u>face</u> <u>(0x100000</u> <u>0f)</u>	<u>Button</u> <u>shadow</u> <u>(0x100000</u> <u>10)</u>	<u>Disabled</u> <u>menu item</u> <u>(0x100000</u> <u>11)</u>
<u>Button</u> <u>highlight</u> <u>(0x100000</u>		

12)

WAV audio playback

The demos in this section use very short and quite noisy samples to keep file size at a minimum.

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;THSndPlaySound(qchPath~`THWAVS.L
ZH'~`SAY1.WAV'~0)}System modal playback (plays once, locks system
until .WAV playback has been completed)
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`THSndPlaySound(qchPath~`[A
RCHIVENAME].LZH'~`[FILENAME]'~0)):THMB(hwndContext~`Template text
copied to clipboard.'~`Template copied'~64)}
THSndPlaySound(qchPath,`[ARCHIVENAME].LZH',`[FILENAME]',0)
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;THSndPlaySound(qchPath~`THWAVS.L
ZH'~`SAY1.WAV'~1)}Normal playback; return control of Windows to user before
completing playback
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`THSndPlaySound(qchPath~`[A
RCHIVENAME].LZH'~`[FILENAME]'~1)):THMB(hwndContext~`Template text
copied to clipboard.'~`Template copied'~64)}
THSndPlaySound(qchPath,`[ARCHIVENAME].LZH',`[FILENAME]',1)
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;THSndPlaySound(qchPath~`THWAVS.L
ZH'~`SAY1.WAV'~9)}Looped WAV audio playback (Click here to stop the
playback)
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`THSndPlaySound(qchPath~`[A
RCHIVENAME].LZH'~`[FILENAME]'~9)):THMB(hwndContext~`Template text
copied to clipboard. Paste into your source document and edit to suit your
project. Remember to remove square brackets.'~`Template copied'~64)}
THSndPlaySound(qchPath,`[ARCHIVENAME].LZH',`[FILENAME]',9)
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;THSndPlaySound(qchPath~`THWAVS.L
ZH'~`SAY1.WAV'~1):THSndPlaySound(qchPath~`THWAVS.LZH'~`SAY1.WAV'~1):T
HSndPlaySound(qchPath~`THWAVS.LZH'~`SAY1.WAV'~1):THSndPlaySound(qch
Path~`THWAVS.LZH'~`SAY1.WAV'~1):PI(`~`anisampa')}WAV audio with
concurrent animation (plays WAV with simultaneous popup of animation...no
template)
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;THSndPlaySound(qchPath~`THWAVS.L
ZH'~`SAY1.WAV'~16):THSndPlaySound(qchPath~`THWAVS.LZH'~`SAY2.WAV'~16)
```

:THSndPlaySound(qchPath~`THWAVS.LZH'~`SAY3.WAV'~16):THSndPlaySound(qchPath~`THWAVS.LZH'~`SAY4.WAV'~16))Multiple WAVs played back in sequence (correct method)

{ewl THNHLP,THIN,!}

THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`THSndPlaySound(qchPath~`[ARCHIVENAME].LZH'~`[FILENAME]'~16);THSndPlaySound(qchPath~`[ARCHIVENAME].LZH'~`[FILENAME]'~16);[additional WAV macros as

needed]'):THMB(hwndContext~`Template text copied to clipboard. Paste into your source document and edit to suit your project. Remember to remove square brackets.'~`Template copied'~64))THSndPlaySound(qchPath,

`[ARCHIVENAME].LZH',`[FILENAME]',16); THSndPlaySound(qchPath,`[ARCHIVENAME].LZH',`[FILENAME]',16); [additional WAV playback macros as needed]

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!}

THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;THSndPlaySound(qchPath~`THWAVS.LZH'~`SAY'~0))Multiple WAVs played back in sequence (incorrect method, multiple .WAVs referenced by filemask rather than individually)

Custom macro demos

```
{ewc THNHLP,THIN,!
  THDEMOS.LZH;CLOSBOOK.BMP;0/0/0/0/0/255/0;;JI(`thdemos.hlp>egg'~`ths
  e')}} THShellExecute()
{ewc THNHLP,THIN,!
  THDEMOS.LZH;CLOSBOOK.BMP;0/0/0/0/0/255/0;;JI(`thdemos.hlp>egg'~`thfe
  '')}} THFileExist()
{ewc THNHLP,THIN,!
  THDEMOS.LZH;CLOSBOOK.BMP;0/0/0/0/0/255/0;;JI(`thdemos.hlp>demo'~`th
  mb')}} THMessageBox()
{ewc THNHLP,THIN,!
  THDEMOS.LZH;CLOSBOOK.BMP;0/0/0/0/0/255/0;;JI(`'~`thmw')}}
  THMoveWindow()
{ewc THNHLP,THIN,!
  THDEMOS.LZH;CLOSBOOK.BMP;0/0/0/0/0/255/0;;PI(`'~`regkey')}}
  THTestLicenseKey()
{ewc THNHLP,THIN,!
  THDEMOS.LZH;CLOSBOOK.BMP;0/0/0/0/0/255/0;;PI(`'~`thstc')}}
  THStringToClipboard()
```

```
{ewr THNHLP,THIN,!  
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th  
demos.hlp>main'~`return')'~`CloseWindow(`'))}THShellExecute()  
demo
```

{ewl THNHLP,THIN,!THDEMOS.LZH;HLP-
OUT.BMP;0/0/2/0;;THShellExecute(hwndApp~`open'~`THINHELP.HTM'~`'~`~1)}This demo will
execute a launch of an HTML file on your hard disk into your browser. This is the HTML file
used as the Internet resource for ThinHelp users. In order for this to work, you'll need to make
sure that your browser isn't currently open. You'll have to watch for similarly troublesome
conditions which you make known to your user when authoring this macro into your own
projects.

```
{ewl THNHLP,THIN,!  
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`THShellExecute(hwndApp~`[pri  
nt|  
open]'~`[AppToLaunch]'~`[FileToLoad]'~`[WorkingDirectoryPath]~`[DisplayState]')  
'~`[AppToLaunch]'~`[FileToLoad]'~`[WorkingDirectoryPath]~`[DisplayState]')  
');THMB(hwndContext~`Template text copied to clipboard. Paste into your source  
document and edit to suit your project. Remember to remove square  
brackets.'~`Template copied'~64)}THShellExecute(hwndApp,`[print|open]',  
`[AppToLaunch]',`[FileToLoad]',`[WorkingDirectoryPath],[DisplayState]')
```

```
{ewr THNHLP,THIN,!
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th
demos.hlp>main'~`return')'~`CloseWindow(`'))}THFileExist() demo
```

{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;IfThenElse(THFileExist(`thdemos.hlp')~`
PI(`~`buttonpop')'~`THMessageBox(hwndContext~`File THDEMOS.HLP not found
in Windows path'~`File not found'~48'))}This demo will hunt for a copy of
THDEMOS.HLP (this file) anywhere in your current Windows path. If it finds it
(and of course it will), it will jump to the dummy popup topic. If it doesn't find
one, it will pop up a "file not found" message box. (See the messagebox demos
for more on this feature.) THFileExists' functionally is identical to the Win95 32
bit Help FileExists() macro.

{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;IfThenElse(THFileExist(`glux9734.txt')~`
PI(`~`buttonpop')'~`THMessageBox(hwndContext~`File GLUX9734.TXT not found
in Windows path'~`File not found'~48'))}This demo will perform the same action,
but instead look for GLUX9734.TXT which is *highly* unlikely to exist on your
system.

{ewl THNHLP,THIN,!
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`IfThenElse(THFileExist(`[FILEN
AME]')~`[Macro()]'~`[Macro()]')):THMB(hwndContext~`Template text copied to
clipboard. Paste into your source document and edit to suit your project.
Remember to remove square brackets.'~`Template copied'~64))
IfThenElse(THFileExist(`[FILENAME]'),`[Macro()]',`[Macro()]')

```
{ewr THNHLP,THIN,!
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark('demo')~`JI('th
demos.hlp>main'~`return')'~`CloseWindow(''))}THMessageBox()
demos
```

Simple message boxes

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;THMessageBox(hwndContext~`Simple
OK messagebox with no icon'~`Insert caption bar text here'~0)}Simple message
box, no icon
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`THMessageBox(hwndContext~`
[MessageContentText]'~`[CaptionBarText]'~0)):THMB(hwndContext~`Template
text copied to clipboard. Paste into your source document and edit to suit your
project. Remember to remove square brackets.'~`Template copied'~64)}
THMessageBox(hwndContext,`[MessageContentText]',`[CaptionBarText]',0)
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;THMessageBox(hwndContext~`Simple
OK messagebox with STOP icon'~`Insert caption bar text here'~16)}Simple
message box with STOP icon
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`THMessageBox(hwndContext~`
[MessageContentText]'~`[CaptionBarText]'~0)):THMB(hwndContext~`Template
text copied to clipboard. Paste into your source document and edit to suit your
project. Remember to remove square brackets.'~`Template copied'~64)}
THMessageBox(hwndContext,`[MessageContentText]',`[CaptionBarText]',16)
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;THMessageBox(hwndContext~`Simple
OK messagebox with question mark icon'~`Insert caption bar text here'~32)}
Simple message box with question mark icon
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`THMessageBox(hwndContext~`
[MessageContentText]'~`[CaptionBarText]'~0)):THMB(hwndContext~`Template
text copied to clipboard. Paste into your source document and edit to suit your
project. Remember to remove square brackets.'~`Template copied'~32)}
THMessageBox(hwndContext,`[MessageContentText]',`[CaptionBarText]',32)
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;THMessageBox(hwndContext~`Simple
OK messagebox with exclamation mark icon'~`Insert caption bar text here'~48)}
Simple message box with exclamation mark icon
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`THMessageBox(hwndContext~`
```

```
[MessageContentText]'~`[CaptionBarText]'~0'):THMB(hwndContext~`Template
text copied to clipboard. Paste into your source document and edit to suit your
project. Remember to remove square brackets.'~`Template copied'~48))
THMessageBox(hwndContext,`[MessageContentText]',`[CaptionBarText]',48)

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;THMessageBox(hwndContext~`Simple
OK messagebox with information icon'~`Insert caption bar text here'~64))Simple
message box with information icon

{ewl THNHLP,THIN,!
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`THMessageBox(hwndContext~`
[MessageContentText]'~`[CaptionBarText]'~0'):THMB(hwndContext~`Template
text copied to clipboard. Paste into your source document and edit to suit your
project. Remember to remove square brackets.'~`Template copied'~64))
THMessageBox(hwndContext,`[MessageContentText]',`[CaptionBarText]',64)
```

Message boxes producing return values

These message boxes produce return values; clicking the appropriate button produces the result specified in the macro.

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;IfThenElse(THMB(hwndContext~`Do
you want to see the dummy popup?'~`MessageBox
demonstration'~4)~`PI('~`buttonpop')~`THMB(hwndContext~`Okay...not
displaying dummy popup'~`Nooo problem...'~64'))Simple Yes/No message box,
no icon

{ewl THNHLP,THIN,!
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`IfThenElse(THMessageBox(hwn
dContext~`[MessageContentText]'~`[CaptionBarText]'~4):`[Macro()]~`[Macro()]')
):THMB(hwndContext~`Template text copied to clipboard. Paste into your source
document and edit to suit your project. Remember to remove square
brackets.'~`Template copied'~64))IfThenElse(THMessageBox(hwndContext,
`[MessageContentText]',`[CaptionBarText]',4);`[Macro()],`[Macro()]')
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;IfThenElse(THMessageBox(hwndConte
xt~`Do you want to see the dummy popup?'~`MessageBox demonstration
'~36)~`PI('~`buttonpop')~`THMB(hwndContext~`Okay...not displaying dummy
popup'~`Nooo problem...'~64'))Simple Yes/No message box, question mark icon

{ewl THNHLP,THIN,!
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`IfThenElse(THMessageBox(hwn
dContext~`[MessageQuestionText]'~`[CaptionBarText]'~36):`[Macro()]~`[Macro()]
'))):THMB(hwndContext~`Template text copied to clipboard. Paste into your
source document and edit to suit your project. Remember to remove square
brackets.'~`Template copied'~64))IfThenElse(THMessageBox(hwndContext,
`[MessageQuestionText]',`[CaptionBarText]',36);`[Macro()],`[Macro()]')
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```



```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;IfThen(THMessageBox(hwndContext~`
About to display the dummy popup. Press CANCEL to abort.'~`MessageBox
demonstration '~65)~`PI('~`buttonpop'))}Simple OK/Cancel message box,
information icon
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`IfThen(THMessageBox(hwndCo
ntext~`[MessageContentText]~`[CaptionBarText]~65):[Macro()]')':THMB(hwndCo
ntext~`Template text copied to clipboard. Paste into your source document and
edit to suit your project. Remember to remove square brackets.'~`Template
copied'~64))IfThen(THMessageBox(hwndContext,[MessageContentText]',
[CaptionBarText]',65);[Macro()]')
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;IfThen(THMessageBox(hwndContext~`
Unable to execute the action requested. Please insure that the software is
correctly installed or press RETRY to try again.'~`MessageBox demonstration
'~21)~`PI('~`buttonpop'))}Simple Retry/Cancel message box, STOP icon. This
demo displays a sample message you could use if a file in the Help/Viewer
system is missing or if conditions aren't correct for the desired action.
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`IfThen(THMessageBox(hwndCo
ntext~`[MessageContentText]~`[CaptionBarText]~21):[Macro()]')':THMB(hwndCo
ntext~`Template text copied to clipboard. Paste into your source document and
edit to suit your project. Remember to remove square brackets.'~`Template
copied'~64))IfThen(THMessageBox(hwndContext,[MessageContentText]',
[CaptionBarText]',21);[Macro()]')
```

“Difficult” message boxes

Three-option message box implementations are designed primarily for non-Help applications. They are not recommended for Help.

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;IfThen(THMessageBox(hwndContext~`
Do you want to see the dummy popup?'~`MessageBox demonstration
'~35)~`PI('~`buttonpop'))}Simple Yes/No/Cancel message box, question mark
icon. No and Cancel produce the same result when used in this IfThen() demo,
which may or may not confuse the end user.
```

```
{ewr THNHLP,THIN,!  
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th  
demos.hlp>main'~`return')'~`CloseWindow(`'))}THMoveWindow()  
demos
```

{ewl THNHLP,THIN,!THDEMOS.LZH;COMPASS.BMP;0/0/2/0;;PW(71~31~880~990~1~`main')}
This demo will resize the current window a number of times while it shows you its tricks. This button is supplied to restore the window to its original coordinates when you're finished examining the effects.

```
{ewl THNHLP,THIN,!  
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;THMoveWindow(hwndApp~10~200~475  
~475~1)}Resizes this window to 475x475 pixels and positions the window 10  
pixels from the top and 200 pixels down from the left.
```

```
{ewl THNHLP,THIN,!  
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`THMovewindow(hwndApp~[Pix  
elsFromLeft]~[PixelsFromTop]~[WidthInPixels]~[HeightInPixels]~1')):THMB(hwnd  
Context~`Template text copied to clipboard. Paste into your source document  
and edit to suit your project. Remember to remove square brackets.'~`Template  
copied'~64)}THMovewindow(hwndApp,[PixelsFromLeft],[PixelsFromTop],  
[WidthInPixels],[HeightInPixels],1)
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

```
{ewl THNHLP,THIN,!  
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;THMoveWindow(hwndApp~-1~-1~475~4  
75~1)}Use the "reorient" button at the top of this topic prior to running this  
demo. Resizes this window to 475x475 pixels but does not move it from its  
present screen position.
```

```
{ewl THNHLP,THIN,!  
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`THMovewindow(hwndApp~-1~-  
1~[WidthInPixels]~[HeightInPixels]~1')):THMB(hwndContext~`Template text  
copied to clipboard. Paste into your source document and edit to suit your  
project. Remember to remove square brackets.'~`Template copied'~64)}  
THMovewindow(hwndApp,-1,-1,[WidthInPixels],[HeightInPixels],1)
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

```
{ewl THNHLP,THIN,!  
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;THMoveWindow(hwndApp~50~50~-1~-1  
~1)}Use the "reorient" button at the top of this topic prior to running this demo.  
Repositions this window 50 pixels in from the left and 50 pixels down from the  
top but does not resize it from its present size.
```

```
{ewl THNHLP,THIN,!  
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`THMovewindow(hwndApp~[Pix  
elsFromLeft]~[PixelsFromTop]~-1~-1~1')):THMB(hwndContext~`Template text  
copied to clipboard. Paste into your source document and edit to suit your  
project. Remember to remove square brackets.'~`Template copied'~64)}  
THMovewindow(hwndApp,[PixelsFromLeft],[PixelsFromTop],-1,-1,1)
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

**{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;THMoveWindow(hwndContext~200~200
~475~475~1)} Resizes only the scrolling area of this window to 475x475 pixels
and positions the text content 200 pixels from the top and 200 pixels in from the
left. (This demo will restore screen coordinates to normal before executing the
resize.) This special effect must be used with care. Use the Back button to re-
enter this demo menu after executing this demo.**

**{ewl THNHLP,THIN,!
THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`THMovewindow(hwndContext~[
PixelsFromLeft]~[PixelsFromTop]~[WidthInPixels]~[HeightInPixels]~1)'):THMB(h
wndContext~`Template text copied to clipboard. Paste into your source
document and edit to suit your project. Remember to remove square
brackets.'~`Template copied'~64)}THMovewindow(hwndContext,
[PixelsFromLeft],[PixelsFromTop],[WidthInPixels],[HeightInPixels],1)**

Demonstrated repeatedly
throughout this demo file.
No additional demonstrations
are presented here.

No demos are provided for this feature. Proper comparative demonstration of license key testing would require intruding on your configuration with key entries which would only be needed for this demonstration and would also require alterations of key values which ThinHelp cannot make on its own. The ThinHelp registry information could be demonstrated but no method of comparative testing is possible using this key value as a reference.

Window sizing control

```
{ewl THNHLP,THIN,!  
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI('THDEMOS.HLP>OTHRHELP'~`wmsi  
zeb')}}Window sizing using the THWaterMark() sizing parameters
```

```
{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI('~`thmw')}}  
Window sizing using the THMoveWindow() macro
```

```
{ewr THNHLP,THIN,!
THDEMOS.LZH;BACK.BMP;0/0/2/0;;IfThenElse(IsMark(`demo')~`JI(`th
demos.hlp>main'~`return')'~`CloseWindow(`'))}Window sizing using
THWaterMark() sizing
```

Show popup again

200X200

200X250

200X300

200X350

200X400

200X450

250X200

250X250

250X300

250X350

250X400

250X450

300X200

300X250

300X300

300X350

300X400

300X450

350X200

350X250

350X300

350X350

350X400

350X450

400X200

400X250

400X300

400X350

400X400

400X450

450X300

450X350

450X400

450X450

500X300

500X350

500X400

500X450

550X300

550X350

550X400

550X450

600X300

600X350

600X400

600X450

This topic shows off a wide range of useful all-purpose sizes using THWaterMark()'s window sizing parameters (the second and third parameters of the THWaterMark() macro).

You will insure more consistent results if you specify a size for the desired window in the [WINDOWS] section which you are sure will be *larger* than the desired size for the window you wish to size at the lowest likely screen resolution (usually 640x480).

Be very careful of window sizing for 640x480 displays. Several 450-pixel-high bitmaps have been included to demonstrate how discourteous an excessively large window can be. To experience the proper effect, it is recommended that you browse this topic in 640x480 large font mode.

This demo is purposely limited to window widths of 600 pixels and we recommend applying the same limit to your authoring unless you know your user will be working in other than 640x480 display mode.

Paning (unsupported)

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;EP(`winhelp.exe panedemo.hlp'~0)}Embedded pane in a helpfile (this launches the stand-alone PANEDEMO.HLP file in a new instance of WinHelp)

Dummy popup used to demonstrate embedded window macros. These might typically be used to discuss trivial features or hide “easter eggs”.

About to launch Help to RTF..

All files in ThinHelp's help system include complete, uncrippled functionality with our Help to RTF conversion software for allowing you to create Word-compatible documentation from helpfiles including all bitmaps and embedded window objects. This feature is reserved exclusively for helpfiles included with Herd Software Development software products. For complete functionality with all helpfiles you will need to own a license to Help to RTF Standard or Help to RTF Pro.

You can use the evaluation version of Help to RTF included in this package to create printable documentation from any of the included helpfiles and optionally load the appropriate helpfile directly into Help to RTF by selecting the **Print Some or All Topics...** item from each helpfile's **F**ile menu.

Select one of the following:

Launch Help to RTF

Learn more about Help to RTF

Standard embedded window bitmaps

{ewl THNHLP,THIN,!THDEMOS.LZH;CAUTION.BMP;0/0/0/0/255/0} **Caution:** do not copy template text directly from the topic window. Use the provided **Copy** buttons. The visible text on your screen has been specially formatted and will not paste properly into your RTF source documents.

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`standardademo')}} Ordinary left-aligned embedded window bitmap

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('ï½bml [filename.bmp (or shg)]ï½'):THMB(hwndContext~`Template text copied to clipboard. Paste into your source document and edit to suit your project. Remember to remove square brackets.'~`Template copied'~64)}} Template text: { bml filename.bmp/shg}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`standardbdemo')}} Ordinary right-aligned embedded window bitmap

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('ï½bmr [filename.bmp(or shg)]ï½'):THMB(hwndContext~`Template text copied to clipboard. Paste into your source document and edit to suit your project. Remember to remove square brackets.'~`Template copied'~64)}} Template text: { bmr filename.bmp/shg}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`standardcdemo')}} Ordinary character-aligned embedded window bitmap

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('ï½bmc [filename.bmp(or shg)]ï½'):THMB(hwndContext~`Template text copied to clipboard. Paste into your source document and edit to suit your project. Remember to remove square brackets.'~`Template copied'~64)}} Template text: { bmc filename.bmp/shg}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`standardddemo')}} 256 color embedded window bitmap (compressed and bagged)

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('ï½ew[l r or c] THNHLP~THIN~![ARCHIVENAME]; [FILENAME.EXT]ï½'):THMB(hwndContext~`Template text copied to clipboard. Paste into your source document and edit to suit your project. Remember to remove square brackets.'~`Template copied'~64)}} Template text: {ew[l r or c] THNHLP,THIN,![ARCHIVENAME];[FILENAME.EXT]}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`standardededemo')}} Callout box

```
{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`i½ew[I or  
r] THNHLP~THIN~![ARCHIVENAME];  
[FILENAME.EXT]i½'):THMB(hwndContext~`Template text copied to clipboard.  
Paste into your source document and edit to suit your project. Remember to  
remove square brackets.'~`Template copied'~64))Template text: {ew[I or r]  
THNHLP,THIN,![ARCHIVENAME];[FILENAME.EXT]}
```

Embedded window SHGs

{ewl THNHLP,THIN,!THDEMOS.LZH;CAUTION.BMP;0/0/0/0/255/0}**Caution:** do not copy template text directly from the topic window. Use the provided **Copy** buttons. The visible text on your screen has been specially formatted and will not paste properly into your RTF source documents.

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`shgdem0')}
Standard SHG Help bitmap (visible hotspot)

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('~½bml
[filename].shg½½'):THMB(hwndContext~`Template text copied to clipboard.
Paste into your source document and edit to suit your project. Remember to
remove square brackets.'~`Template copied'~64))Template text: {bml
filename.shg}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`shgdem0b')}
Standard SHG Help mefatile (visible hotspot)

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('~½bml
[filename].shg½½'):THMB(hwndContext~`Template text copied to clipboard.
Paste into your source document and edit to suit your project. Remember to
remove square brackets.'~`Template copied'~64))Template text: {bml
filename.shg}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`shgdema')}
Ordinary embedded window SHG

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('~½ew[I r or
c] THNHLP~THIN~![ARCHIVENAME];
[FILENAME].SHG½½'):THMB(hwndContext~`Template text copied to clipboard.
Paste into your source document and edit to suit your project. Remember to
remove square brackets.'~`Template copied'~64))Template text: {ew[I r or c]
THNHLP,THIN,![ARCHIVENAME];[FILENAME].SHG}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`shgdemb')}
256-color embedded window bitmap-based SHG

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('~½ew[I r or
c] THNHLP~THIN~![ARCHIVENAME];
[FILENAME].SHG½½'):THMB(hwndContext~`Template text copied to clipboard.
Paste into your source document and edit to suit your project. Remember to
remove square brackets.'~`Template copied'~64))Template text: {ew[I r or c]
THNHLP,THIN,![ARCHIVENAME];[FILENAME].SHG}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`shgdemc')}
Embedded-window SHG graphic created by SHED from a Windows metafile
(.WMF)

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('~½ew[I r or

c] THNHLP~THIN~![ARCHIVENAME];
 [FILENAME].SHG;1/2'):THMB(hwndContext~`Template text copied to clipboard.
 Paste into your source document and edit to suit your project. Remember to
 remove square brackets.'~`Template copied'~64))Template text: {ew[l r or c]
 THNHLP,THIN,![ARCHIVENAME];[FILENAME].SHG}
 {ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
 {ewl THNHLP,THIN,
 THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`thdemos.hlp>demo'~`shgdemd')}}
 SHG graphic used as a hotspotted watermark in place of a .BMP + .SEG
 combination

Static JPEG embedded windows

{ewl THNHLP,THIN,!THDEMOS.LZH;CAUTION.BMP;0/0/0/0/255/0}**Caution:** do not copy template text directly from the topic window. Use the provided **Copy** buttons. The visible text on your screen has been specially formatted and will not paste properly into your RTF source documents.

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`jpstata')}
Standard JPEG embedded window (not added to a compressed archive)

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('~½ewc THNHLP~THIN~[FILENAME].JPG;½'):THMB(hwndContext~`Template text copied to clipboard. Paste into your source document and edit to suit your project. Remember to remove square brackets.'~`Template copied'~64))Template text: { ewc THNHLP,THIN,[FILENAME].JPG}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`jpstatb')}
Standard JPEG embedded window (from a compressed archive)

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('~½ewc THNHLP~THIN~[! or ./PATH/][ARCHIVENAME]; [FILENAME].JPG;½'):THMB(hwndContext~`Template text copied to clipboard. Paste into your source document and edit to suit your project. Remember to remove square brackets.'~`Template copied'~64))Template text: {ewc THNHLP,THIN,[! or ./PATH/][ARCHIVENAME];[FILENAME].JPG}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`jpstatc')}
Standard JPEG embedded window cached to hard disk (performance effect only observable on reboot)

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('~½ewc THNHLP~THIN~+[! or ./PATH/][ARCHIVENAME]; [FILENAME].JPG;½'):THMB(hwndContext~`Template text copied to clipboard. Paste into your source document and edit to suit your project. Remember to remove square brackets.'~`Template copied'~64))Template text: {ewc THNHLP,THIN,+[! or ./PATH/][ARCHIVENAME];[FILENAME].JPG}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`jpstatd')}
JPEG embedded window with 256-color palette control (.PAL file) (See above examples for non-palette-controlled display results)

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('~½ew[l r or c] THNHLP~THIN~[! or ./PATH/][ARCHIVENAME]; [PARTIALFILENAME];½'):THMB(hwndContext~`Template text copied to clipboard. Paste into your source document and edit to suit your project. Remember to remove square brackets.'~`Template copied'~64))Template text: { ewc THNHLP,THIN,[! or ./PATH/][ARCHIVENAME];[PARTIALFILENAME]}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,! THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI('thdemos.hlp>othrhel'~`jpstate')}

JPEG embedded window with metafile overlay

```
{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('î½ewc  
THNHLP~THIN~[! or ./PATH/][ARCHIVENAME];  
[PARTIALFILENAME]î½'):THMB(hwndContext~`Template text copied to  
clipboard. Paste into your source document and edit to suit your project.  
Remember to remove square brackets.'~`Template copied'~64))Template text: {  
ewc THNHLP,THIN,[! or ./PATH/][ARCHIVENAME];[PARTIALFILENAME]}
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`jpstatf'))JPEG
embedded window with transparency mask

```
{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('î½ewc  
THNHLP~THIN~[! or ./PATH/][ARCHIVENAME];  
[PARTIALFILENAME]î½'):THMB(hwndContext~`Template text copied to  
clipboard. Paste into your source document and edit to suit your project.  
Remember to remove square brackets.'~`Template copied'~64))Template text: {  
ewc THNHLP,THIN,[! or ./PATH/][ARCHIVENAME];[PARTIALFILENAME]}
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`jpstatg'))
JPEG embedded window with transparency mask and overlay

```
{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('î½ewc  
THNHLP~THIN~[! or ./PATH/][ARCHIVENAME];  
[PARTIALFILENAME]î½'):THMB(hwndContext~`Template text copied to  
clipboard. Paste into your source document and edit to suit your project.  
Remember to remove square brackets.'~`Template copied'~64))Template text: {  
ewc THNHLP,THIN,[! or ./PATH/][ARCHIVENAME];[PARTIALFILENAME]}
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`jpstath'))
JPEG embedded window with transparency mask, overlay and palette control file

```
{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('î½ewc  
THNHLP~THIN~[! or ./PATH/][ARCHIVENAME];  
[PARTIALFILENAME]î½'):THMB(hwndContext~`Template text copied to  
clipboard. Paste into your source document and edit to suit your project.  
Remember to remove square brackets.'~`Template copied'~64))Template text: {  
ewc THNHLP,THIN,[! or ./PATH/][ARCHIVENAME];[PARTIALFILENAME]}
```

Transparent bitmaps

{ewl THNHLP,THIN,!THDEMOS.LZH;CAUTION.BMP;0/0/0/0/255/0}**Caution:** do not copy template text directly from the topic window. Use the provided **Copy** buttons. The visible text on your screen has been specially formatted and will not paste properly into your RTF source documents.

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`stdtransa')}}16 color transparent bitmap

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('ï½ew[l r or c] THNHLP~THIN~[! or .PATH/][ARCHIVENAME];[FILENAME].BMP;0/0/0/0/[R]/[G]/[B]ï½'):THMB(hwndC ontext~`Template text copied to clipboard. Paste into your source document and edit to suit your project. Remember to remove square brackets.'~`Template copied'~64)}}Template text: {ew[l r or c] THNHLP,THIN,[! or .PATH/][ARCHIVENAME];[FILENAME].BMP;0/0/0/0/[R]/[G]/[B]}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`stdtransb')}}256 color transparent bitmap

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('ï½ew[l r or c] THNHLP~THIN~[! or .PATH/][ARCHIVENAME];[FILENAME].BMP;0/0/0/0/[R]/[G]/[B]ï½'):THMB(hwndC ontext~`Template text copied to clipboard. Paste into your source document and edit to suit your project. Remember to remove square brackets.'~`Template copied'~64)}}Template text: {ew[l r or c] THNHLP,THIN,[! or .PATH/][ARCHIVENAME];[FILENAME].BMP;0/0/0/0/[R]/[G]/[B]}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`stdtransc')}}TrueColor transparent JPEG (not usually useful due to difficulty in controlling the transparency mask)

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('ï½ew[l r or c] THNHLP~THIN~[! or .PATH/][ARCHIVENAME];[FILENAME].JPG;[n]/[n]/0/0/[R]/[G]/[B]ï½'):THMB(hwnd Context~`Template text copied to clipboard. Paste into your source document and edit to suit your project. Remember to remove square brackets.'~`Template copied'~64)}}Template text: {ew[l r or c] THNHLP,THIN,[! or .PATH/][ARCHIVENAME];[FILENAME].JPG;[n]/[n]/0/0/[R]/[G]/[B]}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`stdtransd')}}Transparent bitmap with added transparency mask (not supported)

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`stdtranse')}}Transparent bitmap with metafile overlay (not supported)

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI('~`stdtransf')}}Transparent 256-color bitmap with palette control (used in difficult situations)

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`i½ew[l r or
c] THNHLP~THIN~[! or
./PATH/][ARCHIVENAME];[PARTFILENAME];0/0/0/0/[R]/[G]/[B]i½'):THMB(hwndC
ontext~`Template text copied to clipboard. Paste into your source document and
edit to suit your project. Remember to remove square brackets.'~`Template
copied'~64)}}Template text: {ew[l r or c] THNHLP,THIN,[! or ./PATH/
[ARCHIVENAME];[PARTFILENAME];0/0/0/0/[R]/[G]/[B]}

Embedded window buttons

{ewl THNHLP,THIN,!THDEMOS.LZH;CAUTION.BMP;0/0/0/0/255/0} **Caution:** do not copy template text directly from the topic window. Use the provided **Copy** buttons. The visible text on your screen has been specially formatted and will not paste properly into your RTF source documents.

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI( THDEMOS.HLP>EGG'~`bsampa')
Standard 16-color embedded window button
```

```
{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`i½ew[I r or
c] THNHLP~THIN~[! or ./PATH/][ARCHIVENAME];
[FILENAME].BMP;0/0/2i½'):THMB(hwndContext~`Template text copied to
clipboard. Paste into your source document and edit to suit your project.
Remember to remove square brackets.'~`Template copied'~64))Template text: {
ew[I r or c] THNHLP,THIN,[! or ./PATH/][ARCHIVENAME];[FILENAME].BMP;0/0/2}
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI( THDEMOS.HLP>EGG'~`bsampb')
Standard 256-color embedded window button
```

```
{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`i½ew[I r or
c] THNHLP~THIN~[! or ./PATH/][ARCHIVENAME];
[FILENAME].BMP;0/0/2i½'):THMB(hwndContext~`Template text copied to
clipboard. Paste into your source document and edit to suit your project.
Remember to remove square brackets.'~`Template copied'~64))Template text: {
ew[I r or c] THNHLP,THIN,[! or ./PATH/][ARCHIVENAME];[FILENAME].BMP;0/0/2}
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI( THDEMOS.HLP>EGG'~`bsampc')
TrueColor JPEG embedded window button
```

```
{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`i½ew[I r or
c] THNHLP~THIN~[! or
./PATH/][ARCHIVENAME];[FILENAME].BMP;[n]/[n]/2i½'):THMB(hwndContext~`Te
mplate text copied to clipboard. Paste into your source document and edit to suit
your project. Remember to remove square brackets.'~`Template copied'~64))
Template text: {ew[I r or c] THNHLP,THIN,[! or ./PATH/][ARCHIVENAME];
[FILENAME].JPG;[[n]/[n]/2}
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI( THDEMOS.HLP>EGG'~`bsampd')
Transparent embedded window button (not supported)
```

```
{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`i½ew[I r or
c] THNHLP~THIN~[! or
./PATH/][ARCHIVENAME];[FILENAME].BMP;0/0/2/0/[R]/[G]/[B]i½'):THMB(hwndC
ontext~`Template text copied to clipboard. Paste into your source document and
edit to suit your project. Remember to remove square brackets.'~`Template
copied'~64))Template text: {ew[I r or c] THNHLP,THIN,[! or ./PATH/}
```

[ARCHIVENAME];[FILENAME].BMP;0/0/2/0/[R]/[G]/[B]}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!

THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(THDEMOS.HLP>EGG'~`bsampe')}

Animated embedded window button

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('i½ew[l r or
c] THNHLP~THIN~[! or

./PATH/][ARCHIVENAME];[PARTFILENAME];0/0/2/[n]/[R]/[G]/[B]i½'):THMB(hwnd
Context~`Template text copied to clipboard. Paste into your source document
and edit to suit your project. Remember to remove square brackets.'~`Template
copied'~64))Template text: {ew[l r or c] THNHLP,THIN,[! or ./PATH/]

[ARCHIVENAME];[PARTFILENAME];0/0/2/[n]/[R]/[G]/[B]}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!

THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(THDEMOS.HLP>EGG'~`bsampf')}

Buttonized callout box

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC('i½ew[l r or
c] THNHLP~THIN~[! or ./PATH/][ARCHIVENAME];

[FILENAME].BMP;0/0/2i½'):THMB(hwndContext~`Template text copied to
clipboard. Paste into your source document and edit to suit your project.

Remember to remove square brackets.'~`Template copied'~64))Template text: {
ew[l r or c] THNHLP,THIN,[! or ./PATH/][ARCHIVENAME];[FILENAME].BMP;0/0/2}

Animated embedded windows

{ewl THNHLP,THIN,!THDEMOS.LZH;CAUTION.BMP;0/0/0/0/255/0}**Caution:** do not copy template text directly from the topic window. Use the provided **Copy** buttons. The visible text on your screen has been specially formatted and will not paste properly into your RTF source documents.

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI(`~`anisampa')}
Basic animated embedded window

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`i½ew[I r or c] THNHLP~THIN~[! or .PATH/][ARCHIVENAME];[PARTFILENAME];0/0/0/[n]i½'):THMB(hwndContext~`Template text copied to clipboard. Paste into your source document and edit to suit your project. Remember to remove square brackets.'~`Template copied'~64))
Template text: {ew[I r or c] THNHLP,THIN,[! or .PATH/][ARCHIVENAME];[PARTFILENAME];0/0/0/[n]}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI(`~`anisampb')}
Correctly-displayed transparent animation

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`i½ew[I r or c] THNHLP~THIN~[! or .PATH/][ARCHIVENAME];[PARTFILENAME];0/0/0/[n]/[R]/[G]/[B]i½'):THMB(hwndContext~`Template text copied to clipboard. Paste into your source document and edit to suit your project. Remember to remove square brackets.'~`Template copied'~64))
Template text: {ew[I r or c] THNHLP,THIN,[! or .PATH/][ARCHIVENAME];[PARTFILENAME];0/0/0/[n]/[R]/[G]/[B]}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI(`~`anisampc')}
Incorrectly-displayed transparent animation

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`thdemos.hlp>egg'~`bsampe')}
Animated embedded window button

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`i½ew[I r or c] THNHLP~THIN~[! or .PATH/][ARCHIVENAME];[PARTFILENAME];0/0/2/[n]i½'):THMB(hwndContext~`Template text copied to clipboard. Paste into your source document and edit to suit your project. Remember to remove square brackets.'~`Template copied'~64))
Template text: {ew[I r or c] THNHLP,THIN,[! or .PATH/][ARCHIVENAME];[PARTFILENAME];0/0/2/[n]}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`thdemos.hlp>egg'~`anisampd')}
Animation with scroll bar; start when clicked or static and step through with scroll buttons (recommended method)

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`i½ew[I r or c] THNHLP~THIN~[! or .PATH/][ARCHIVENAME];[PARTFILENAME];0/0/2/[n]}

./PATH/][ARCHIVENAME];[PARTFILENAME];0/0/8/[n]i½'):THMB(hwndContext~`T
emplate text copied to clipboard. Paste into your source document and edit to
suit your project. Remember to remove square brackets.'~`Template copied'~64))
Template text: {ew[I r or c] THNHLP,THIN,[! or ./PATH/][ARCHIVENAME];
[PARTFILENAME];0/0/8/[n]}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI(`~`anisampe'')
Animation with metafile overlay on last frame

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`i½ew[I r or
c] THNHLP~THIN~[! or

./PATH/][ARCHIVENAME];[PARTFILENAME];0/0/4/[n]i½'):THMB(hwndContext~`T
emplate text copied to clipboard. Paste into your source document and edit to
suit your project. Remember to remove square brackets.'~`Template copied'~64))
Template text: {ewl r or c] THNHLP,THIN,[! or ./PATH/][ARCHIVENAME];
[PARTFILENAME];0/0/4/[n]}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI(`~`anisampf'')
User-defined animation sequence

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`i½ew[I r or
c] THNHLP~THIN~[! or

./PATH/][ARCHIVENAME];[PARTFILENAME]/[PARTFILENAME]/[ETC.];0/0/0/[n]i½'
) :THMB(hwndContext~`Template text copied to clipboard. Paste into your source
document and edit to suit your project. Remember to remove square
brackets.'~`Template copied'~64))Template text: {ew[I r or c] THNHLP,THIN,[!
or ./PATH/][ARCHIVENAME];[PARTFILENAME]/[PARTFILENAME]/[ETC.];0/0/0/[n]}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`thdemos.hlp>egg'~`anisampg'')}Play-
once animation

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`i½ew[I r or
c] THNHLP~THIN~[! or

./PATH/][ARCHIVENAME];[PARTFILENAME]/[PARTFILENAME]/[ETC.];0/0/1/[n]i½'
) :THMB(hwndContext~`Template text copied to clipboard. Paste into your source
document and edit to suit your project. Remember to remove square
brackets.'~`Template copied'~64))Template text: {ew[I r or c] THNHLP,THIN,[!
or ./PATH/][ARCHIVENAME];[PARTFILENAME]/[PARTFILENAME]/[ETC.];0/0/1/[n]}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`thdemos.hlp>egg'~`anisamph'')}
Start-when-clicked animation

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`i½ew[I r or
c] THNHLP~THIN~[! or

./PATH/][ARCHIVENAME];[PARTFILENAME]/[PARTFILENAME]/[ETC.];0/0/16/
[n]i½'):THMB(hwndContext~`Template text copied to clipboard. Paste into your
source document and edit to suit your project. Remember to remove square

brackets.'~`Template copied'~64)}}Template text: {ew[I r or c] THNHLP,THIN,[!
or
./PATH/][ARCHIVENAME];[PARTFILENAME]/[PARTFILENAME]/[ETC.];0/0/16/[n]}

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`thdemos.hlp>egg'~`anisampi')}
Everything but the kitchen sink (play once, buttonized, scroll bars, metafile on
last frame)

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI(`~`anisampi2')}
Multiple animations at differing speeds

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`thdemos.hlp>demo'~`anisampj')}
User-definable frame advance rate

{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}

{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;THSetAnimationSpeed(600):PI(`~`anisa
mpk')}}Animated JPEG

{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`i½ew[I r or
c] THNHLP~THIN~[! or
./PATH/][ARCHIVENAME];[PARTFILENAME];0/0/0/[n]i½')}:THMB(hwndContext~`T
emplate text copied to clipboard. Paste into your source document and edit to
suit your project. Remember to remove square brackets.'~`Template copied'~64)}}
Template text: {ew[I r or c] THNHLP,THIN,[! or ./PATH/][ARCHIVENAME];
[PARTFILENAME];0/0/4/[n]}

Metafile overlays

{ewl THNHLP,THIN,!THDEMOS.LZH;CAUTION.BMP;0/0/0/0/255/0}**Caution:** do not copy template text directly from the topic window. Use the provided **Copy** buttons. The visible text on your screen has been specially formatted and will not paste properly into your RTF source documents.

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`thdemos.hlp>othrhlp'~`jpstate')}
JPEG embedded window with metafile overlay
```

```
{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`i½ewc
THNHLP~THIN~[! or ./PATH/][ARCHIVENAME];
[PARTIALFILENAME]i½'):THMB(hwndContext~`Template text copied to
clipboard. Paste into your source document and edit to suit your project.
Remember to remove square brackets.'~`Template copied'~64))Template text: {
ewc THNHLP,THIN,[! or ./PATH/][ARCHIVENAME];[PARTIALFILENAME]}
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

```
{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI(`~`stdtranse')}
Transparent bitmap with metafile overlay
```

```
{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`i½ew[I r or
c] THNHLP~THIN~[! or
./PATH/][ARCHIVENAME];[PARTFILENAME];0/0/0/0/[R]/[G]/[B]i½'):THMB(hwndC
ontext~`Template text copied to clipboard. Paste into your source document and
edit to suit your project. Remember to remove square brackets.'~`Template
copied'~64))Template text: {ew[I r or c] THNHLP,THIN,[! or ./PATH/]
[ARCHIVENAME];[PARTFILENAME];0/0/0/0/[R]/[G]/[B]}
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

```
{ewl THNHLP,THIN,!THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;PI(`~`anisampe')}
Animation with metafile overlay on last frame
```

```
{ewl THNHLP,THIN,!THDEMOS.LZH;COPYBUTT.BMP;0/0/2/0;;THSTC(`i½ew[I r or
c] THNHLP~THIN~[! or
./PATH/][ARCHIVENAME];[PARTFILENAME];0/0/4/[n]i½'):THMB(hwndContext~`T
emplate text copied to clipboard. Paste into your source document and edit to
suit your project. Remember to remove square brackets.'~`Template copied'~64))
Template text: {ew[I r or c] THNHLP,THIN,[! or ./PATH/][ARCHIVENAME];
[PARTFILENAME];0/0/4/[n]}
```

```
{ewc THNHLP,THIN,!THDEMOS.LZH;SEPDEMO.BMP}
```

```
{ewl THNHLP,THIN,!
THDEMOS.LZH;DEMOBUTT.BMP;0/0/2/0;;JI(`thdemos>egg'~`anisampi')}
Everything but the kitchen sink (play once, transparent, buttonized, scroll bars,
metafile on last frame)
```

