

## TAnimated Component

Properties

Events

Tasks

Copyright

### Unit

Animate.Pas

### Description

The TAnimated component displays simple animations on the form.











The animation that appears is determined by the BitMap property. The bitmap consists of a series of two or more Frames arranged in filmstrip fashion.

Filmstrip frames may be sequenced automatically at fixed time intervals or may be changed under program control. When sequenced at fixed time intervals, the Reverse property determines the sequence direction and the Loop property determines if sequencing continues indefinitely.

## Item 1

This is the Item 1 topic.

## Properties

-  Key Properties
-  BitMap
-  Frame
-  FrameCount
-  Height
-  Interval
-  Loop
-  Play
-  Reverse
-  TransparentColor
- Visible
- Width

## Using The TAnimated Component

TAnimated Reference

### Tasks

#### To sequence the animation at timed intervals...

Set the Interval property to the desired interval, the Frame property to the starting frame number, and set the Play property to True to initiate sequencing.

- To sequence in the reverse direction, set the Reverse property to True.
- To sequence the filmstrip continuously, set the Loop property to True.
- To synchronize sounds or other events to a particular frame, use the OnFrameChange event and check the Frame property.

#### To sequence the animation under program control...

Set the Interval property to zero. Use the Frame property to specify the frame to be displayed.

## TAnimated Frame Property

### Declaration

```
property Frame: Boolean;
```

### Description

The Frame property indicates or determines which filmstrip frame will be displayed.

### Example:

```
with Film do  
  Frame := 3;
```

will cause frame 3 to be drawn. Frame values are constrained to be within 0 and FrameCount-1, so incrementing or decrementing Frame past the limits will result in wrap-around.

## Reverse Property

### Declaration

**property** Reverse: Boolean;

Setting Reverse to True will cause the Play property to play the filmstrip in reverse order.

## BitMap Property

### Declaration

**property** BitMap: TBitmap;

The BitMap property determines the animation that appears on the TAnimated control. The bitmap is arranged a series of two or more horizontal frames in filmstrip fashion.

## TransparentColor Property

### Declaration

**property** TransparentColor: TColor;

You can designate one of the bitmap's colors to represent a transparent area with the TransparentColor property. Instead of this color being painted, the form's color will show through.

The default value for TransparentColor is -1 indicating no transparent color. The object inspector displays clScrollBar for -1.

Drawing time is increased somewhat when a transparent color is used.



## TAnimated Events



OnChangeFrame

OnDragDrop

OnDragOver

OnEndDrag

OnMouseDown

OnMouseMove

OnMouseUp

## FrameCount Property

### Description

property FrameCount: Integer;

FrameCount indicates how many frames the filmstrip has. The Frame property runs from 0 to FrameCount - 1.

## OnChangeFrame Event

### Declaration

**property** OnChangeFrame: TNotifyEvent;

The OnChangeFrame event occurs when a change is made in the Frame number. This event may be used to synchronize sounds with particular filmstrip frames.

## Loop Property

### Declaration

**property** Loop: Boolean;

The Loop property determines if the filmstrip will Play continuously or terminate when the last frame is displayed.

## Play Property

### Declaration

**property** Play: Boolean;

Once the Interval property has been set, setting Play to True will cause TAnimated to sequence through the frames of the filmstrip. Reverse determines the sequencing direction, and Loop determines whether the sequence loops continuously.

### Example:

```
with MyAnimated do
  begin
    Frame := FrameCount - 1;    {Start at last frame}
    Reverse := True;           {do it backwards}
    Loop := True;               {and continuously}
    Play := True;               {start}
  end;
```

### See Also:

[Frame Property](#)

[Interval Property](#)

[Loop Property](#)

[Reverse Property](#)

## Interval Property

### Declaration

**property** Interval: Integer;

### Description

The Interval property specifies the time (in milliseconds) between frame changes when the filmstrip is Played.

Use an Interval value of 0 (the default) if you plan to program frame changes using the Frame property.

### Note:

Because of the way the system timer operates, actual time intervals are in multiples of 55 milliseconds. Thus values from 1 to 55 will result in a 55 ms. interval, values between 56 and 110 will result in a 110 ms. interval, etc.

## Filmstrip Layout

The bitmap used by the TAnimated component consists of a series of frames which normally are sequenced from left to right. The total length of the bitmap is exactly equal to the FrameCount property times the Width property.



## Suggestions

Start by selecting a frame width and height. Estimate the number of frames required and make the total bitmap width exactly equal to the product of the width and framecount. The total width may be modified later if more or fewer frames are required.

Many animations can be accomplished by copying the same object for each frame and modifying each one in place. If the object is not to move, it's essential that the distance between objects be precisely equal to the frame width.

If it's necessary to change the bitmap width or height, be sure to **uncheck** the option which expands or shrinks the bitmap to fit the new size.

I've found that Borland's Resource Workshop works well for these bitmaps. The Delphi Image Editor may also be used but its 1024 limit for bitmap width might be overly restrictive for some situations.

**© Copyright 1995 by L. David Baldwin**

**All Rights Reserved**

There is no charge for the use of the TAnimated component. However, if you come up with some interesting animations, I'd enjoy seeing them.

This package may be copied and distributed freely providing that it is not modified, no fee is charged, and it is not made part of a package for which a charge is made. If you upload this package to other bulletin boards, I'd appreciate it if you would try to keep the upload current.

Please report all bugs, suggestions, and problems to:

Dave Baldwin,  
CompuServe ID: 76327,53.

22 Fox Den Rd., (Summer)  
Hollis, NH 03049  
(603) 465-7857

144 13th St. East, (Winter)  
Tierra Verde, FL 33715  
(813) 867-3030



## Height Property

### Declaration

**property** Height: Integer;

### Description

The Height property determines the vertical size of a component or object. For TAnimated, the Height property is set automatically when a bitmap is loaded.



