

## PlayEnhMetaFile

The **PlayEnhMetaFile** function displays the picture stored in the specified enhanced-format metafile.

```
BOOL PlayEnhMetaFile(  
    HDC hdc,           // handle of a device context  
    HENHMETAFILE hemf, // handle of an enhanced metafile  
    CONST RECT * lpRect // address of bounding rectangle  
);
```

### Parameters

*hdc*

Identifies the device context for the output device on which the picture will appear.

*hemf*

Identifies the enhanced metafile.

*lpRect*

Points to a [RECT](#) structure that contains the coordinates of the bounding rectangle used to display the picture. The coordinates are specified in logical units.

### Return Value

If the function succeeds, the return value is TRUE.

If the function fails, the return value is FALSE. To get extended error information, call [GetLastError](#).

### Remarks

When an application calls the **PlayEnhMetaFile** function, Windows uses the picture frame in the enhanced-metafile header to map the picture onto the rectangle pointed to by the *lpRect* parameter. (This picture may be sheared or rotated by setting the world transform in the output device before calling **PlayEnhMetaFile**.) Points along the edges of the rectangle are included in the picture.

An enhanced-metafile picture can be clipped by defining the clipping region in the output device before playing the enhanced metafile.

If an enhanced metafile contains an optional palette, an application can achieve consistent colors by setting up a color palette on the output device before calling **PlayEnhMetaFile**. To retrieve the optional palette, use the [GetEnhMetaFilePaletteEntries](#) function.

An enhanced metafile can be embedded in a newly created enhanced metafile by calling **PlayEnhMetaFile** and playing the source enhanced metafile into the device context for the new enhanced metafile.

The states of the output device context are preserved by this function. Any object created but not deleted in the enhanced metafile is deleted by this function.

To stop this function, an application can call the **CancelDC** function from another thread to terminate the operation. In this case, the function returns FALSE.