

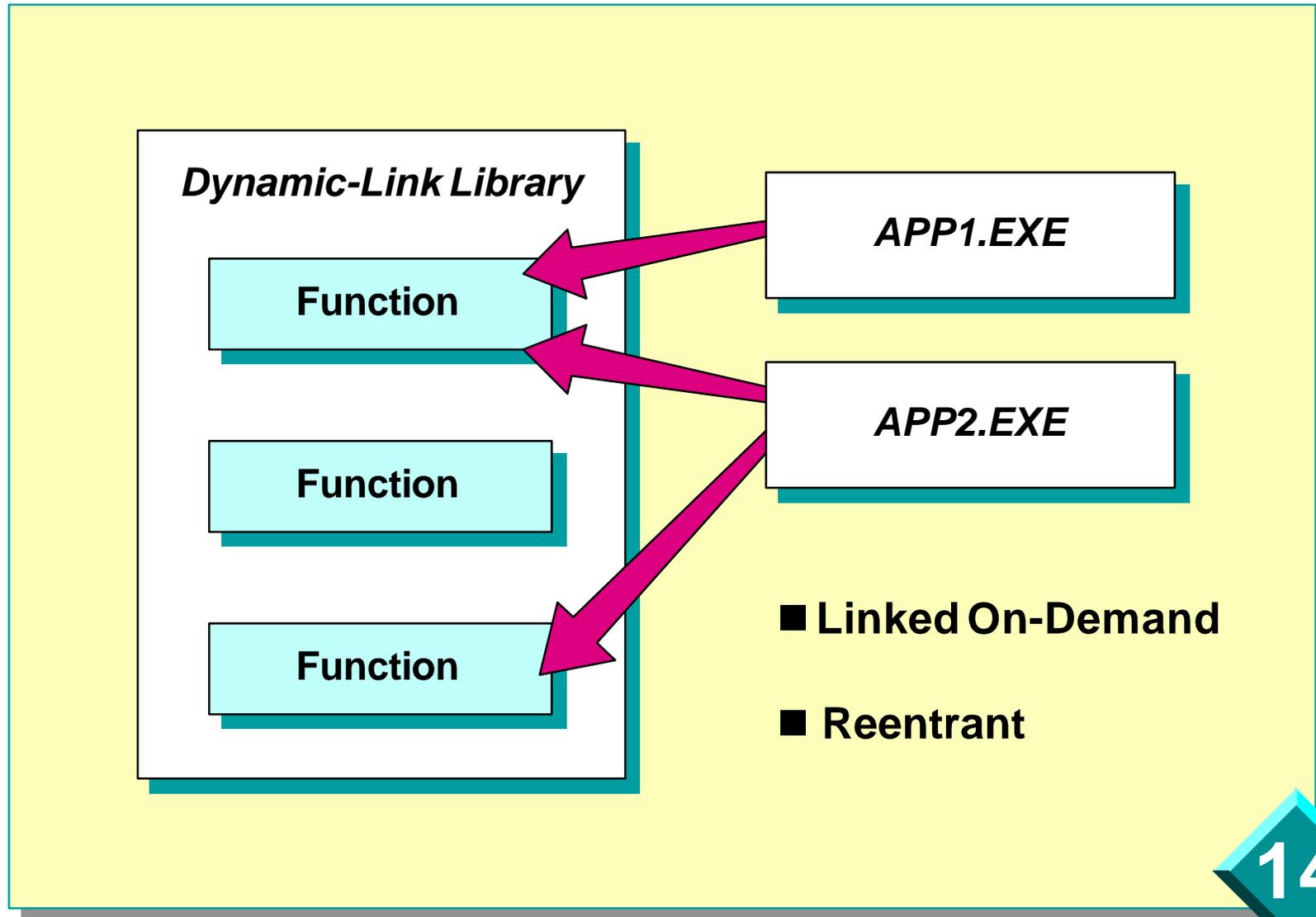
Using Dynamic-Link Libraries

14

◆ Overview

- What Are Dynamic-Link Libraries (DLLs)?
- Implementing DLLs in a Visual Basic Application
 - Declaring DLLs
 - Calling DLL functions
- Examples of Using DLLs

What Are DLLs?



What DLL Functions Can I Call?

- Windows DLLs
 - **User** - Manipulates Windows; Interface Devices
 - **GDI** - Renders Graphics on Screens, Printers
 - **Kernel** - Does O/S Stuff; Tasks, Memory, I/O
 - About 800 Functions in Windows 3.1
- *myown*.DLL - FORTRAN, COBOL, C/C++, PASCAL
- Other Applications (DLLs)
- VBRUN300.DLL

Information on Windows API Functions

- WIN31API.HLP
- WIN31WH.HLP
- WIN31EXT.TXT
- WIN30API.TXT

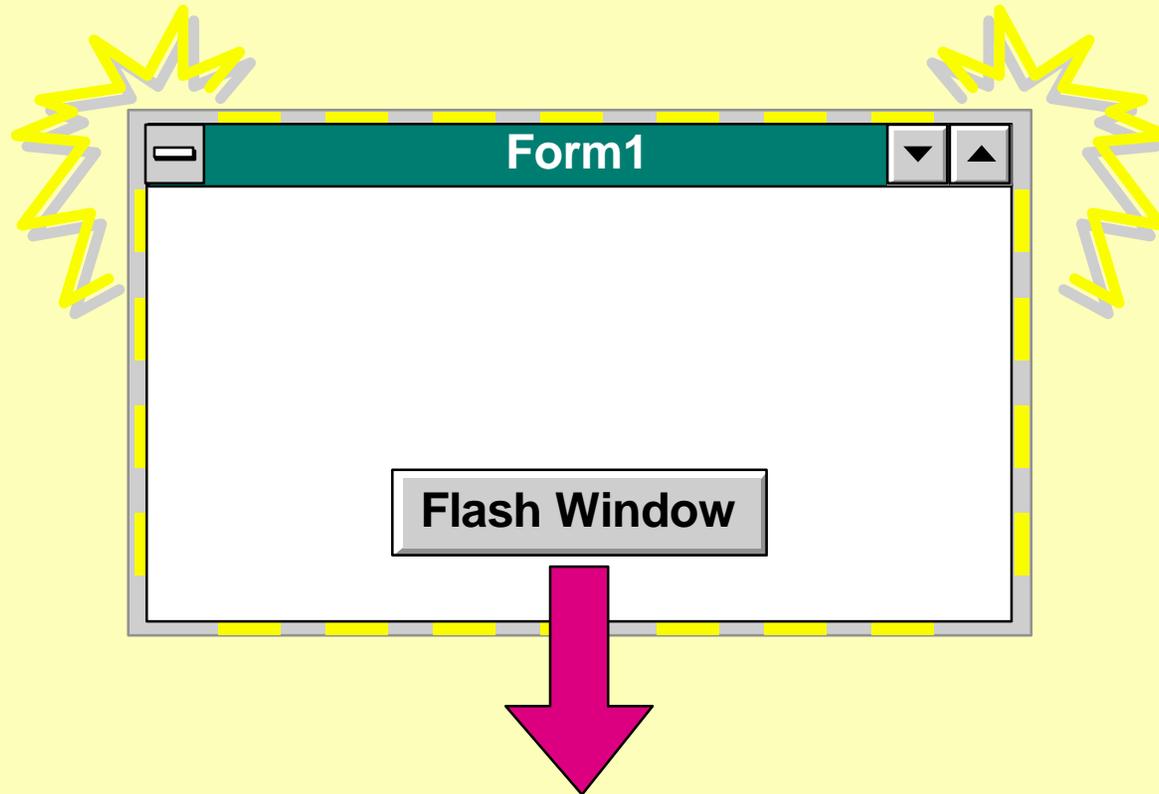
Steps to Implement DLLs in a Visual Basic Application

- 1 Declare the DLL Function
- 2 Write the Call Statement for the DLL Function
- 3 *Save Your Work !!!*
- 4 Test the Application

Run-Time Properties for DLL's

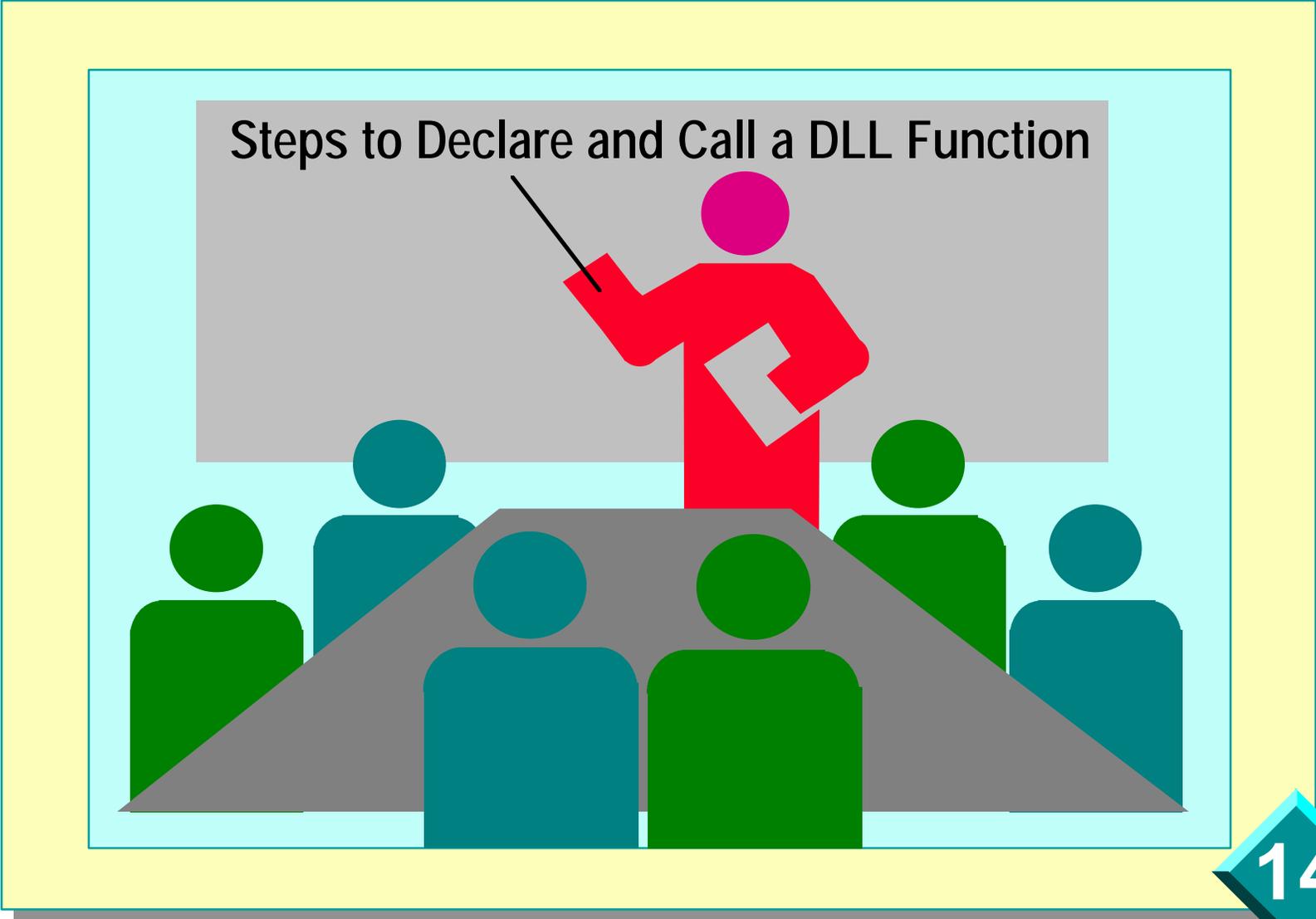
- **hWnd - Window Handle**
 - All Objects (Controls and Forms)
- **hDC - Device Context Handle**
 - Forms
 - PictureBox
 - Printer

A Simple Example

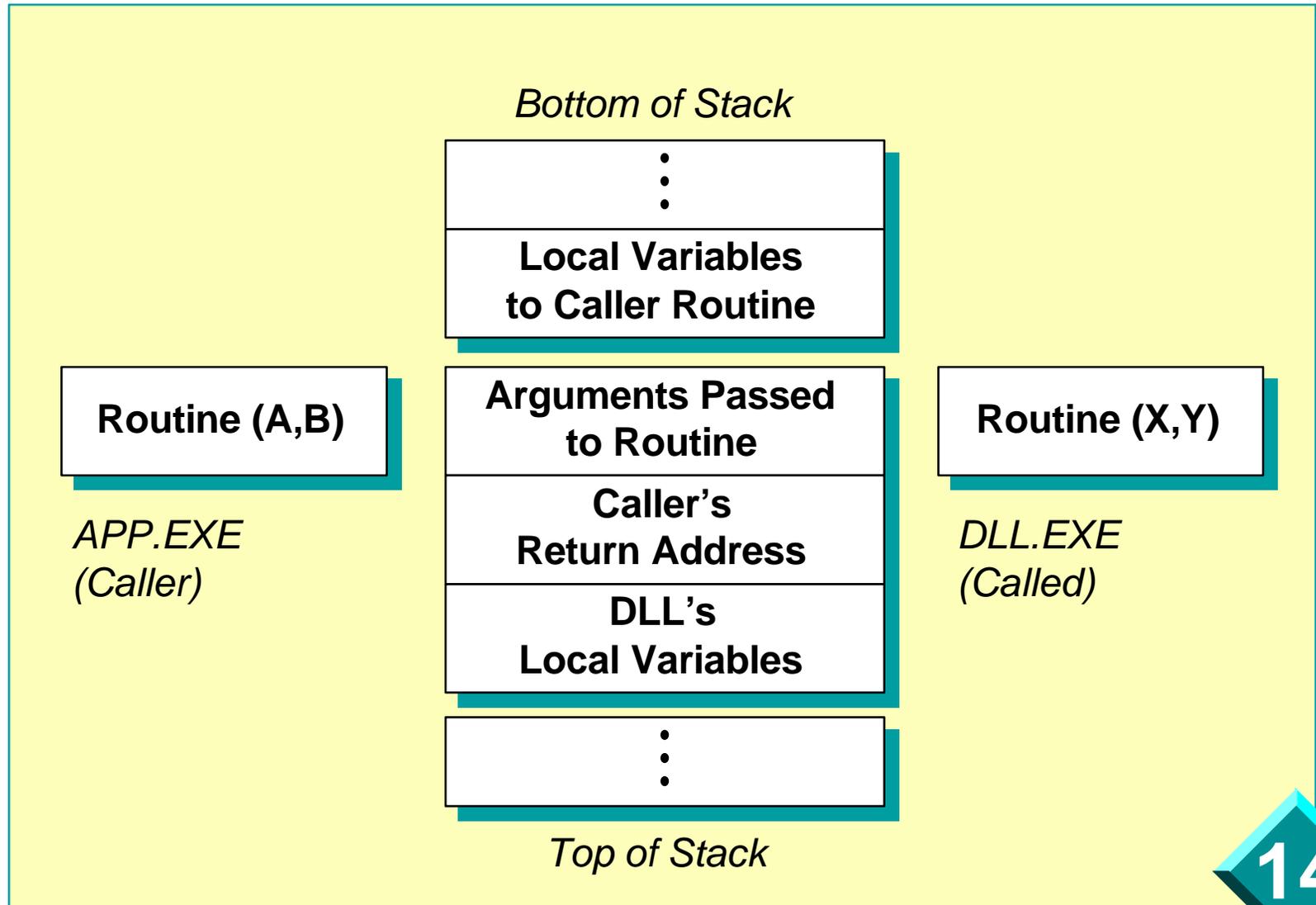


```
result = FlashWindow(Form1.hWnd, 1)
```

Using FlashWindow—Walkthrough



What Happens When You Call a Function in a DLL?



Declare the Function

If Function Returns NULL

```
Declare Sub BringWindowToTop Lib "User"  
↳ (ByVal hWnd As Integer)
```

If Function Returns Value

```
Declare Function IsZoomed Lib "User"  
↳ (ByVal hWnd As Integer) As Integer
```

If Function has "Illegal" VB Syntax Name

```
Declare Function lopen Lib "Kernel"  
↳ Alias "_lopen" (ByVal lpPathName As String,  
↳ ByVal iReadWrite As Integer) As Integer
```

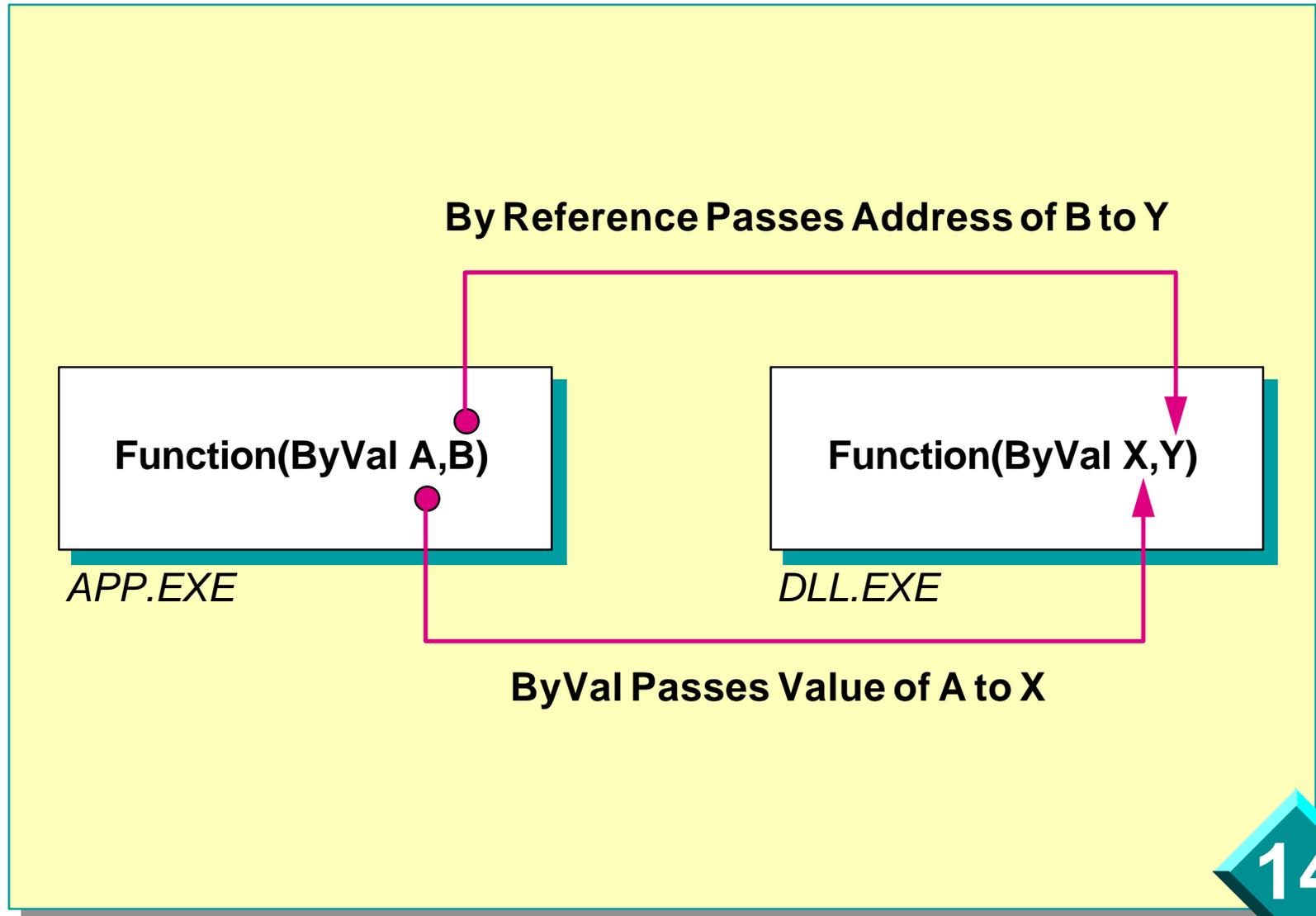
You Must Pass the Correct Data Types

Size	C	Visual Basic	Visual Basic abbreviation
1 byte	char	(notes)	n/a
2 bytes	int	Integer	%
4 bytes	long	Long	&
4 bytes	float	Single	!
8 bytes	double	Double	#
8 bytes	n/a	Currency	@

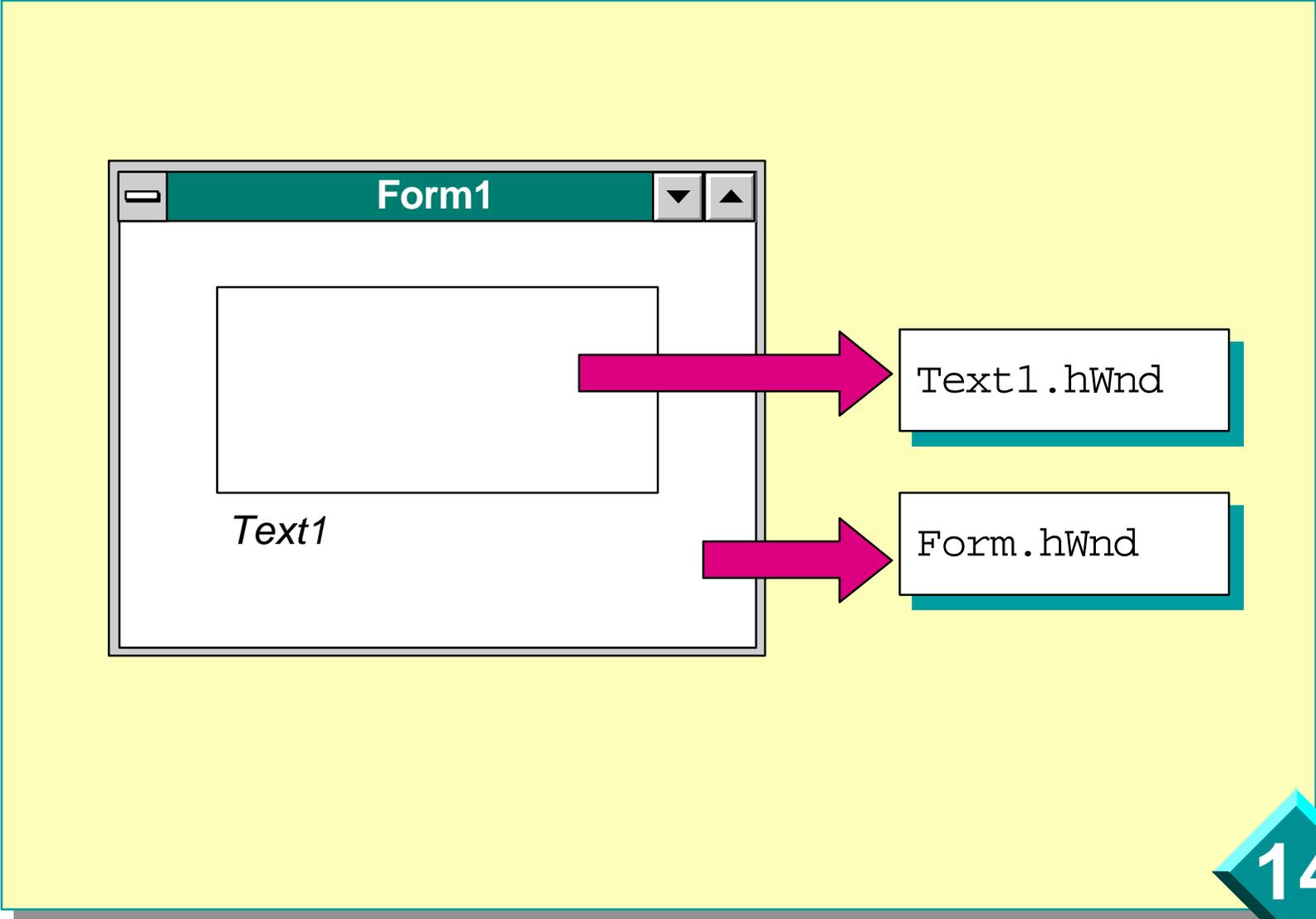
Note: Arguments Aligned on Word (16 bit) Boundaries

14

Arguments Can Be Passed by Value or by Reference



Getting the Windows Handle of an Object

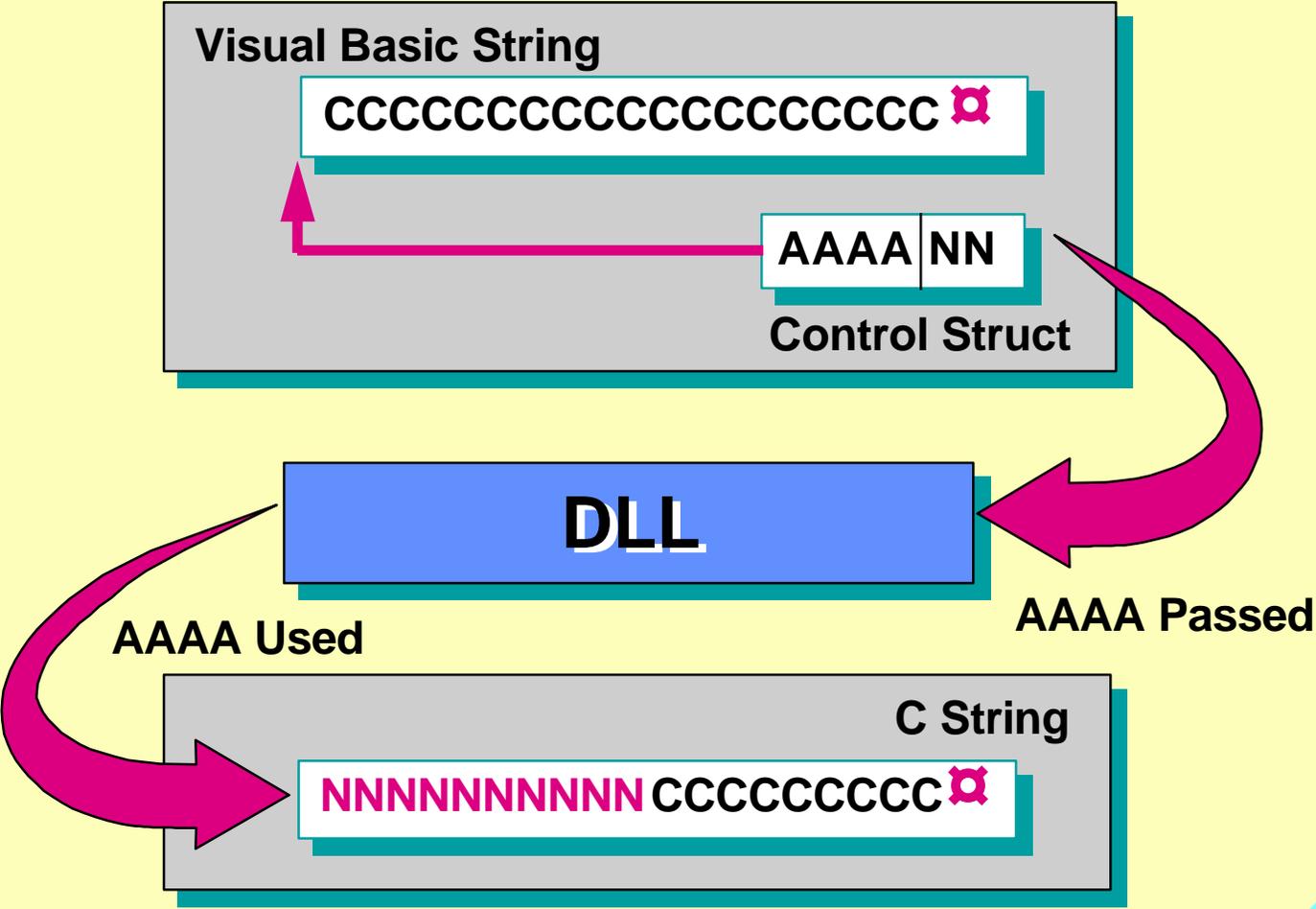


Calling DLL Functions That Pass Properties

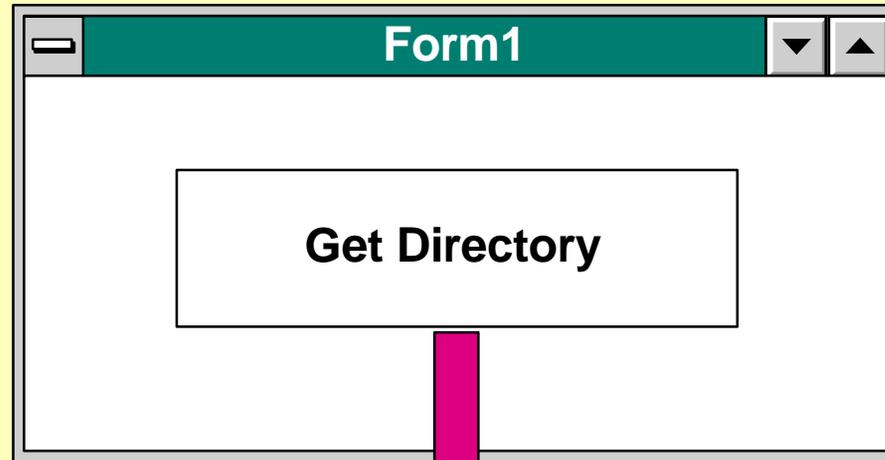
- Always Pass Properties by Value
- This Includes String Variables
- ByVal Strings Auto-Converted by VB to Address

How Strings Are Passed

Declaration or Call - Pass ByVal



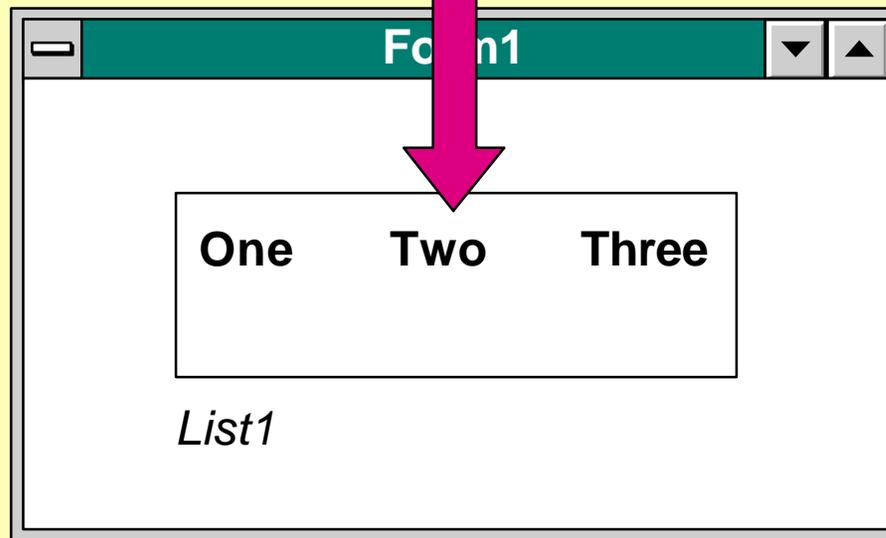
Passing Strings - Have Enough Space



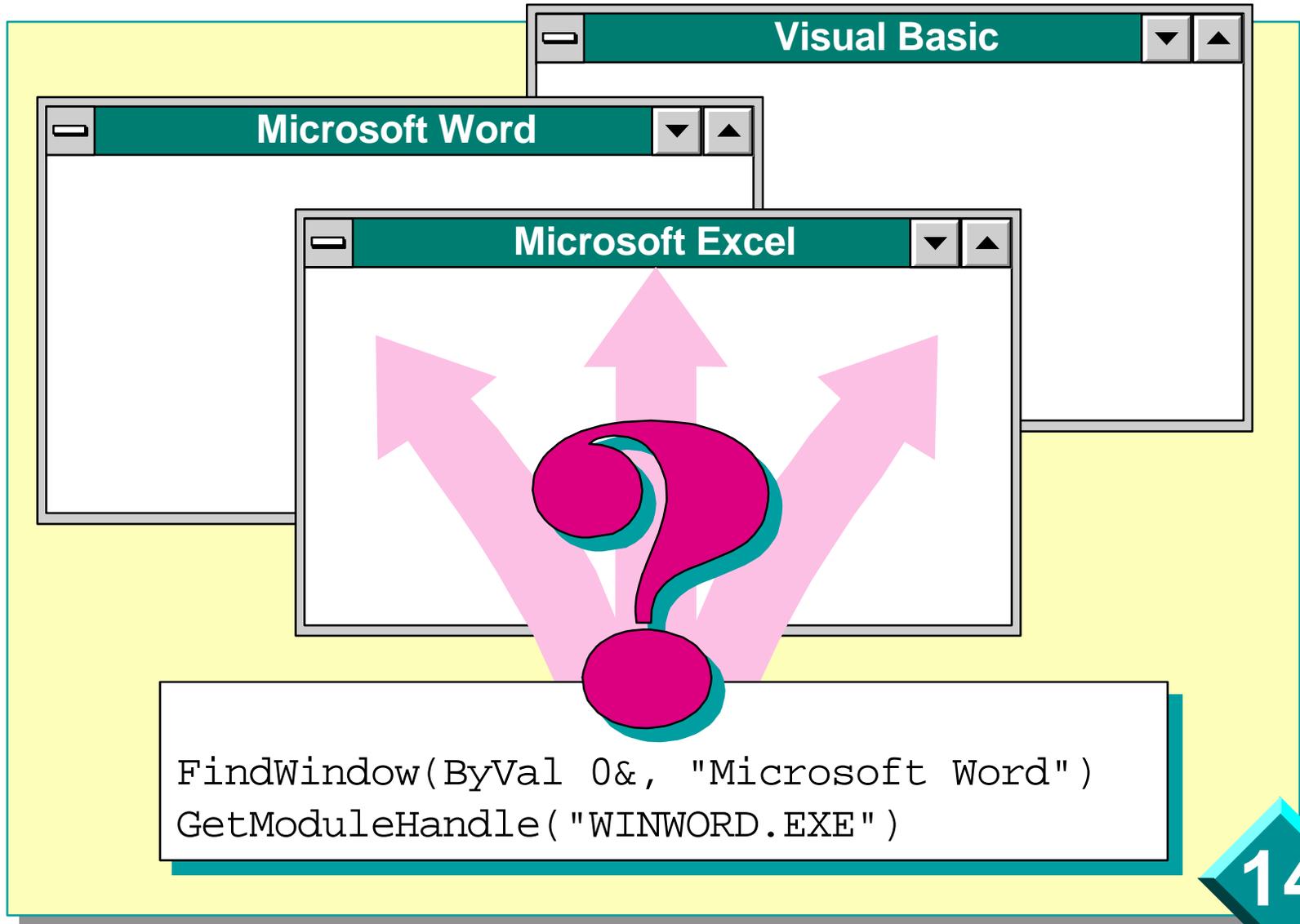
```
Dim FilePath As String *255  
GetWindowsDirectory (FilePath$...)
```

Set Tab Stops in a List Box - SendMessage

```
...LB_SETTABSTOPS, 3, tabs(1)
```



Check Whether an Application Is Running

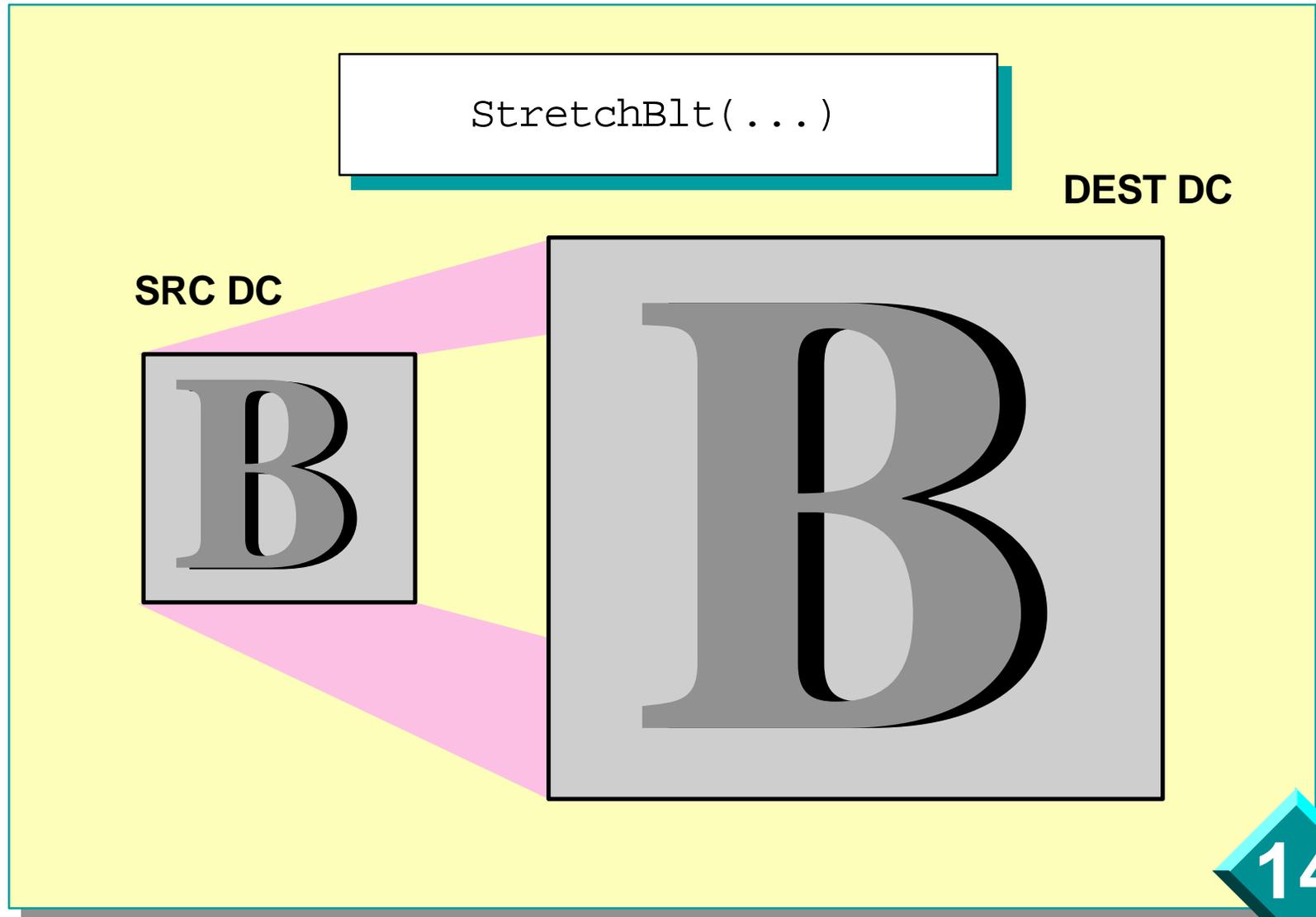


```
FindWindow(ByVal 0&, "Microsoft Word")  
GetModuleHandle("WINWORD.EXE")
```

Determining When a Process Has Finished

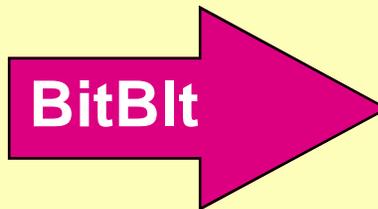
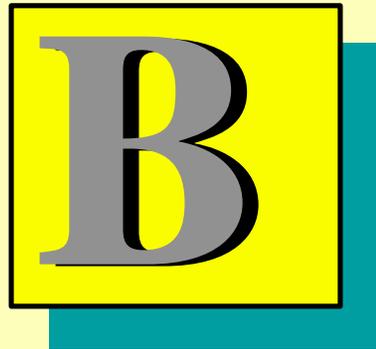
- Start a Process Using the **Shell** Function
- Call **GetActiveWindow**
- Call **IsWindow** in a Loop
- Include **DoEvents** Statement to Yield CPU

StretchBlt Sizes a Bitmap

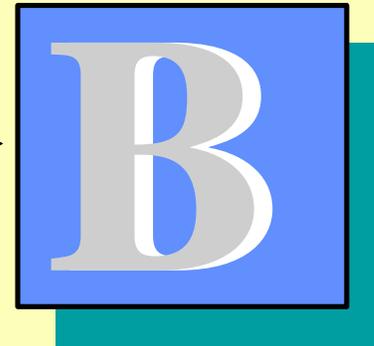


Using BitBlt to Copy a Picture

SRC DC



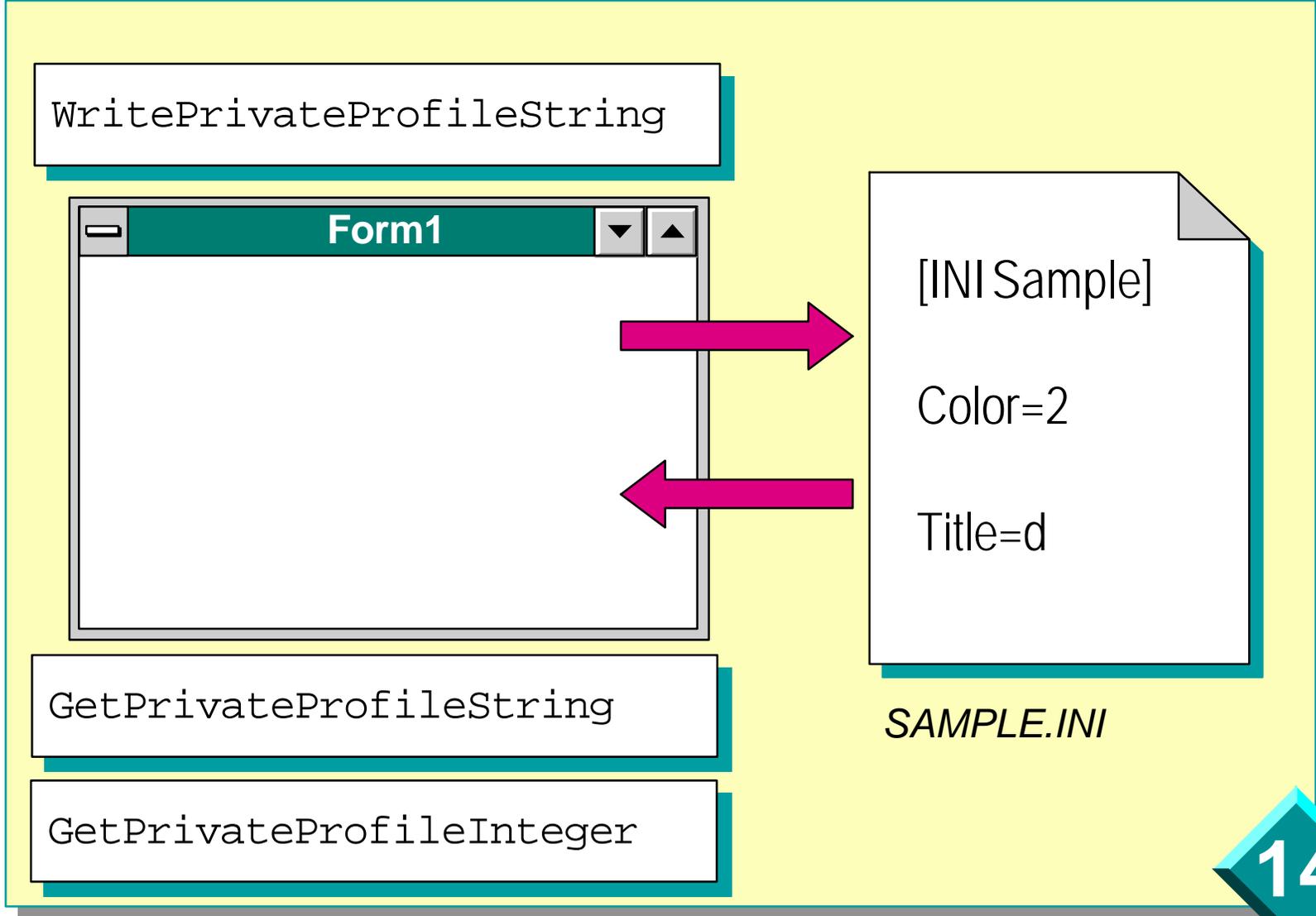
DEST DC



dwRop Parameter Determines Type of Copy
Direct, Inverted, XOr, And, Or, etc.

dwRop = SRCINVERT

Creating an APP.INI File



Summary

- What Are Dynamic-Link Libraries (DLLs)?
- Implementing DLLs in a Visual Basic Application
 - Declaring DLLs
 - Calling DLL functions
- Examples of Using DLLs