Visual C++

Presented by: Jigme Tenzing
CIS510 User Interface Programming
What will be Covered?

- C++ & Windows Programming
- Programming Windows UI
- Visual C++ Express IDE
- Widgets in Visual C++
- Layout Managers in Visual C++
- Event Handling in Visual C++
- Conclusion
C++ & Windows Programming

- Compiled language
- Uses Win32 API for UI programming on the Microsoft Windows platform
- Wrappers:
  - Microsoft Foundation Classes
  - C++/CLI (Common Language Infrastructure) is Microsoft’s language specification intended to supersede Managed Extensions for C++ to allow C++ code to be targeted to the CLR for the .Net Framework
Win32: Windows Programming

- Every Windows application has the WinMain function

```c
int WINAPI WinMain(
    HINSTANCE hInstance,
    HINSTANCE hPrevInstance,
    LPSTR lpCmdLine,
    int nCmdShow;
```
Win32: Windows Programming

- The main Event loop:

```c
while (GetMessage(&msg, NULL, 0, 0)) {
    TranslateMessage(&msg);
    DispatchMessage(&msg);
}
```
The WinProc callback function

LRESULT CALLBACK WndProc(HWND hWnd, UINT message, WPARAM wParam, LPARAM lParam) {
    PAINTSTRUCT ps;
    HDC hdc;
    switch (message) {
        case WM_PAINT:
            hdc = BeginPaint(hWnd, &ps);
            EndPaint(hWnd, &ps);
            break;
        case WM_DESTROY:
            PostQuitMessage(0);
            break;
        default:
            return DefWindowProc(hWnd, message, wParam, lParam);
            break;
    }
    return 0;
}
While(GetMessage()){
    Translate(&msg);
    Dispatch(&msg);
}

WinProc(){
    Switch()
    {
        case WM_Paint:
            TextOut("Hello World");
            break;
        case WM_Destroy:
            Application::quit;
            break;
    }
}
Visual C++ Express

- Freely available at:
  http://www.microsoft.com/express/vc/
Provides an integrated environment for developing applications
- Coding environment
- Source management
- Compiling and Debugging
- GUI builder
Compiling and Debugging

Source management

Coding Environment
Drag and Drop objects

Form Designer and source editor

Object Properties & Events
Demo
Widgets: Win32 API

- Every Graphical Object is a window
- Desktop is root window
- Application window: WinMain
- Child windows, Controls & Menus
  - Child Windows arrange the client space of the main window.
  - Buttons, Edit, Static, Combo box, scroll bar, Date and time picker
  - Menus can only be added to non child windows
Win32 API uses Child windows to arrange controls in the client space of the main window. But the Common Language Infrastructure (C++/CLI) provides higher level layout managers:

- FlowLayoutPanel
  - Arranges from left to right
- TableLayoutPanel
  - We can specify number of columns and rows
  - Arrange objects inside the cells
Event Handling

- In the Win32 API, whenever a window is created the operating system assigns a block of memory for information related to the window including the address of the window proc which handles all events for that window.

- The Applications WinProc functions has to handle all events:
  - WM_Paint: draw, redraw events
  - WM_Timer: time based events
  - WM_Destroy: exit window events
  - WM_Command: command events
  - WM_LBUTTONDOWNDBCLK: left mouse button double clicked
  - etc.
WM_COMMAND: further processed

WM_COMMAND wNotifyCode = HIWORD(wParam);
wl D = LOWORD(wParam);
hwndCtl = (HWND) lParam;

wNotifyCode: specifies whether it is from a control or an accelerator.

wl D: specifies the source Identifier

hwndCtl: it provides a handle to the control or is NULL otherwise
Conclusion

- Compare with Java
- Compare with Tcl/Tk