



Next step for dxdiag: defining the dialog for each tab, adding text

Dan Kegel <dank@kegel.com>

Thu, Feb 3, 2011 at 6:28 PM

Reply-To: ucla-cs130-wine@googlegroups.com

To: ucla-cs130-wine@googlegroups.com

I think a useful next step for dxdiag would be for everyone to take a look at how the 2009 project handled putting static text on the system tab. See <http://code.google.com/p/yadxdiag/source/browse/trunk/2009/src/system.c>

For each item, there are two lines, one to get a handle to the text field,
 hWnd = GetDlgItem(hTabDisplay, SYSTEMTAB_TIMEDATA);
 and another to set its contents,
 SendMessage(hWnd, WM_SETTEXT, 0, (LPARAM)systemInfo.timeLocalized);

Where does that text field get defined? In the resource file, which defines a dialog for each tab, like this:

```
SYSTEMTAB_DLG DIALOGEX 10, 19, 448, 246
STYLE WS_CHILD | WS_VISIBLE | WS_TABSTOP
FONT 8, "MS Shell Dlg"
{
  LTEXT "This tool reports detailed information about the DirectX \
components and drivers installed on your system. It lets you test \
functionality, diagnose problems, and change your system configuration to \
work best. \012\012 If you know what area is causing the problem, click the \
appropriate tab above. Otherwise, you can use the Next Page button below to \
visit each page in sequence. \012\012 The More Help page lists some other \
tools that may help with the problem you are experiencing.",
SYSTEMTAB_DESCRIPTION, 0, 0, 448, 65
  GROUPBOX "System Information", SYSTEMTAB_INFO, 0, 65, 448, 181
  RTEXT "Current Date/Time:", SYSTEMTAB_TIMELABEL, 10, 85, 150, 10
  LTEXT "Unknown", SYSTEMTAB_TIMEDATA, 170, 85, 250, 10
  RTEXT "Computer Name:", SYSTEMTAB_NAMELABEL, 10, 97, 150, 10
  LTEXT "Unknown", SYSTEMTAB_NAMEDATA, 170, 97, 250, 10
  ...
}
```

Where is that dialog referred to in the C code? In a function

```
/* Function to update the currently displayed tab */
BOOL DxDiag_SelectTab(UINT i)
{
  /* Destroy the current tab */
  if (hTabDisplay != NULL) DestroyWindow(hTabDisplay);

  /* Create the appropriate tab to display */
  switch (i) {
    case 0:
      hTabDisplay = CreateDialog(hInstance,
```

```
MAKEINTRESOURCE(SYSTEMTAB_DLG),
hTabCtrl, System_DlgProc);
```

```
System_DisplayData();
break;
```

```
...
```

in main.c, which is called from the main window's dialog procedure:

```
LRESULT CALLBACK DxDiag_DlgProc(HWND hWnd, UINT uMsg, WPARAM wParam,
LPARAM lParam)
{
    int i;
    LPNMHDR pnmh;

    /* Process messages */
    switch(uMsg) {
        case WM_INITDIALOG:
            DxDiag_CreateTabCtrl(hWnd);
            return TRUE;

        case WM_NOTIFY:
            pnmh = (LPNMHDR)lParam;
            switch(pnmh->code) {
                case TCN_SELCHANGE:
                    i = TabCtrl_GetCurSel(pnmh->hwndFrom);
                    DxDiag_SelectTab(i);
            }
    }
    ...
}
```

(The main window in the 2009 version is a dialog, which is a specialized kind of window; presumably that was easier somehow. Both dialogs and windows have callback procedures where you handle various window messages.)

So, for tomorrow, I'd like to see each team add its dialog to the resource file, and get it created in main.c when their tab is active. You can add the data fields later, but at least the static labels should be there.

- Dan

--

You received this message because you are subscribed to the Google Groups "ucla-cs130-wine" group. To post to this group, send email to ucla-cs130-wine@googlegroups.com. To unsubscribe from this group, send email to ucla-cs130-wine+unsubscribe@googlegroups.com. For more options, visit this group at <http://groups.google.com/group/ucla-cs130-wine?hl=en>.