Converting from traditional to simplified Chinese, part 3: Highlighting differences

devblogs.microsoft.com/oldnewthing/20050713-12

July 13, 2005

Raymond Chen

One of the things that is interesting to me as a student of the Chinese languages is to recognize where the traditional and simplified Chinese scripts differ. Since this is my program, I'm going to hard-code the color for simplified Chinese script: maroon.

To accomplish the highlighting, we take advantage of listview's custom-draw feature. Custom-draw allows you to make minor changes to the way items are displayed on the screen. It's a middle ground between having listview do all the work (via default drawing behavior) and having the program do all the work (via owner-draw).

The custom-draw cycle for shell common controls consists of series of `NM_CUSTOMDRAW` notifications, starting with the most general and getting more specific. The reason for the break-down is multi-fold. First, it allows the listview control to short-circuit a portion of custom-draw behavior if the parent window does not indicate that it wishes to customize a particular behavior. This reduces message traffic and improves performance when large numbers of items are being drawn. Second, it allows the parent window to target its customizations to the drawing stages it is interested in.

Listviews are peculiar among the shell common controls in that its items sometimes (but not always) have sub-items. This complicates the drawing process since it requires listview to accomodate both styles: large icon view does not use sub-items, but report view does. To address this, the `CDDS_ITEMPREPAINT` stage is entered when an item is about to paint, and any changes made by the parent window are considered to be effective for the entire item. If you want to make changes on a per-subitem basis, you have to respond to `CDDS_ITEMPREPAINT | CDDS_SUBITEM` and set your properties (or reset them if you want to return to the default) for that sub-item.

With those preliminary remarks settled, we can dive in.

```
class RootWindow : public Window
{
 …
protected:
 …
 LRESULT OnLVCustomDraw(NMLVCUSTOMDRAW* pcd);
 …
private:
 HWND m_hwndLV;
 COLORREF m_clrTextNormal;
 Dictionary m_dict;
};
```

We declare our listview custom-draw handler as well as the member variable in which we remember the normal text color so that we can reset it for columns we do not intend to colorize.

```
LRESULT RootWindow::OnNotify(NMHDR *pnm)
{
 switch (pnm->code) {
 case LVN_GETDISPINFO:
  OnGetDispInfo(CONTAINING_RECORD(pnm, NMLVDISPINFO, hdr));
  break;
 case NM_CUSTOMDRAW:
  if (pnm->hwndFrom == m_hwndLV) {
   return OnLVCustomDraw(CONTAINING_RECORD(
                         CONTAINING_RECORD(pnm, NMCUSTOMDRAW, hdr),
                                              NMLVCUSTOMDRAW, nmcd));
  }
  break;
 }
 return 0;
}
```

If we receive a `NM_CUSTOMDRAW` notification from the listview control, we call our new handler. The multiple calls to the `CONTAINING_RECORD` macro are necessary because the `NMHDR` structure is nestled two levels deep inside the `NMLVCUSTOMDRAW` structure.

```
LRESULT RootWindow::OnLVCustomDraw(NMLVCUSTOMDRAW* pcd)
{
 switch (pcd->nmcd.dwDrawStage) {
 case CDDS_PREPAINT: return CDRF_NOTIFYITEMDRAW;
 case CDDS_ITEMPREPAINT:
  m_clrTextNormal = pcd->clrText;
  return CDRF_NOTIFYSUBITEMDRAW;
 case CDDS_ITEMPREPAINT | CDDS_SUBITEM:
  pcd->clrText = m_clrTextNormal;
  if (pcd->iSubItem == COL_SIMP &&
    pcd->nmcd.dwItemSpec < (DWORD)Length()) {
    const DictionaryEntry& de = Item(pcd->nmcd.dwItemSpec);
    if (de.m_pszSimp) {
      pcd->clrText = RGB(0x80, 0x00, 0x00);
    }
  }
  break;
 }
 return CDRF_DODEFAULT;
}
```

During the `CDDS_PREPAINT` stage, we indicate our desire to receive `CDDS_ITEMPREPAINT` notifications. During the `CDDS_ITEMPREPAINT` stage, we save the normal text color and indicate that we want to receive sub-item notifications. It is in the sub-item notification `CDDS_ITEMPREPAINT | CDDS_SUBITEM` that the real work happens.

First, we reset the color to the default on the assumption that we will not need to colorize this column. But if the column is the simplified Chinese column, if the item number is valid, and if the simplified Chinese is different from the traditional Chinese, then we set the text color to maroon.

That's enough with the Chinese/English dictionary for now. All this time, and we don't even have search capability yet! We'll work on that next month.

Raymond Chen

Follow