

dbEditEx 技術報告

DBFWIN / WINDOW 滑鼠支援與光棒配色

專案：STK 庫存管理系統 (Harbour GTWVT Console Mode)

日期：2026-04-17

文件：dbedit_mouse.prg / color_theme.prg / WINFORM.PRG

一、背景與目標

STK 庫存管理系統從 Clipper 轉換到 Harbour GTWVT Console Mode。原系統使用 DBEDIT 進行資料瀏覽，轉換後需要：

- 加入完整的滑鼠支援（點擊、滾輪、hover、右鍵）
- DBFWIN 和 WINDOW 模組的瀏覽光棒改為整列高亮（Row-Wide）
- 光棒配色整合到 color_theme.prg 統一管理
- PgDn/PgUp 翻頁功能修正

1.1 涉及的檔案

檔案	角色
dbedit_mouse.prg	dbEditEx() 函式 — 取代原生 DBEDIT
mouse_handler.prg	MsHandler CLASS — MsBrowseKey() 統一滑鼠處理
DBFWIN.PRG	DBF 視窗瀏覽模組（使用 dbEditEx）
WINDOW.PRG	視窗瀏覽模組（使用 dbEditEx）
TABLE.PRG	表格瀏覽模組（直接使用 TBrowse）
WINFORM.PRG	FancyColors() — TBrowse 配色函式
color_theme.prg	LoadBrowseHiColors() — 光棒配色管理
SYSTEM.CH	BROWSE_ROW_COLOR / BROWSE_CELL_COLOR 定義
STK.PRG	aColor[] 陣列初始化

二、dbEditEx 架構

2.1 為什麼用 dbEditEx 取代 DBEDIT

Harbour 內建的 DBEDIT 有以下限制：

- 過濾 K_MOUSEMOVE — 無法實作 hover 變色
- C53 compat 模式消化 K_LDBLCLK (1006) — callback 收不到雙擊
- 無法直接控制 TBrowse 的 colorRect — 光棒只能高亮單一 cell

dbEditEx 是基於 Harbour 原始碼重寫的 DBEDIT 替代函式，內部使用 TBrowse 物件，完全相容原有的 callback 介面，同時新增滑鼠事件處理。

2.2 主迴圈架構

```
DO WHILE IContinue
/* 1. Stabilize: 繪製/更新 TBrowse */
DO WHILE .T.
    nKey := Inkey()
    IF oBrowse:stabilize() ; EXIT ; ENDIF
ENDDO

/* 2. Idle: 呼叫 callback (DE_IDLE) */
CallUser( oBrowse, xUserFunc, 0, ... )
oBrowse:forceStable()

/* 3. Highlight: 整列高亮 */
colorRect row {2,2} + cell {8,8}

/* 4. Wait: MsBrowseKey() 等待按鍵/滑鼠 */
nKey := MsBrowseKey( 0, oBrowse )

/* 5. Restore: 還原光棒 */
colorRect {1,1}

/* 6. Process: 處理按鍵 (SWITCH) */
SWITCH nKey ...
```

ENDDO

三、滑鼠事件處理

3.1 MsBrowseKey() — 統一滑鼠入口

所有 browse 模組的 InKey 等待都改用 MsBrowseKey() (MsHandler CLASS 的 METHOD) 。它處理以下滑鼠事件：

事件	處理方式	說明
K_MOUSEMOVE	hover 變色	底部功能提示 + Row0 公司名
K_LBUTTONDOWN	MsHitTest/click	browse 內=選行, 外=功能提示
K_RBUTTONDOWN	hb_keyPut(K_ESC)	右鍵=ESC 離開
K_MWFORWARD	oBrowse:up()	滾輪向上
K_MWBACKWARD	oBrowse:down()	滾輪向下
1006 (DbClick)	hb_keyPut(K_F10)	雙擊=F10 修改

3.2 dbEditEx 內建滑鼠處理

dbEditEx 的 SWITCH 區塊處理以下滑鼠事件 (在 MsBrowseKey 之外)：

- K_LBUTTONDOWN → TBMouse() 移動游標到點擊位置
- K_LDBLCLK → TBMouse() + CallUser(K_LDBLCLK)
- K_RBUTTONDOWN → CallUser(K_RBUTTONDOWN)
- K_MWFORWARD/BACKWARD → 滾輪上下 (含 browse 區域判斷)

3.3 PgDn/PgUp 修正

原始 dbEditEx 的 K_PGDN/K_PGUP 有兩個 bug：

- 使用 oBrowse:down()/up() 只移一行，應為 pageDown()/pageUp()
- 多餘的 MRow()/MCol() 滑鼠位置判斷 (鍵盤事件不需要)

/* 修正前 */

CASE K_PGDN

IF MRow() >= ... ; oBrowse:down() ; ENDIF

/* 修正後 */

CASE K_PGDN

oBrowse:pageDown()

四、整列高亮 (Row-Wide Highlight)

4.1 TBrowse 顏色系統原理

TBrowse 使用 colorSpec 字串定義所有可用顏色，每個色位用逗號分隔：

```
colorSpec = "pos1, pos2, pos3, pos4, pos5, pos6, pos7, pos8"
      |   |   |   |           |
      base row col colHi      cell
```

每個欄位有 defColor 指定平時用哪兩個色位：

```
defColor := {3, 4}
      | |
      | +--- selected 用 colorSpec[4]
      +----- normal 用 colorSpec[3]
```

colorRect 暫時替換指定矩形內所有 cell 的 defColor：

```
colorRect({rowPos, 1, rowPos, colCount}, {2, 2})
= 把整列的 defColor 從 {3,4} 換成 {2,2}
= 全部用 colorSpec[2] ( 列高亮色 ) 顯示
```

4.2 三段式 colorRect (row & cell 合體)

參考 TABLE.PRG 的做法，將整列分三段上色：

```
/* 1. 當前格左邊 = 列高亮色 */
colorRect({rowPos, 1, rowPos, colPos-1}, {2, 2})

/* 2. 當前格右邊 = 列高亮色 */
colorRect({rowPos, colPos+1, rowPos, colCount}, {2, 2})

/* 3. 當前格 = cell 高亮色 ( 不同色區分 ) */
colorRect({rowPos, colPos, rowPos, colPos}, {8, 8})

/* 等待按鍵 */
nKey := MsBrowseKey( 0, oBrowse )

/* 還原 */
```

```
colorRect({rowPos, 1, rowPos, colCount}, {1, 1})
```

4.3 失敗經驗與成功關鍵

嘗試 1：直接 colorRect (失敗)

只用 colorRect({rowPos, 1, rowPos, colCount}, {2, 2})，部分欄位無法上色。原因：colorSpec 只有 5 位，position 8 不存在；defColor 用預設 {1,2}，colorRect 改動不明顯。

嘗試 2：colorSpec 暫換重繪 (失敗)

暫時將 colorSpec 全改成光棒色，refreshCurrent + forceStable 重繪。但 forceStable 重繪所有可見行，導致全部行變成光棒色。

嘗試 3：螢幕層直接繪製 (被否決)

用 hb_ScrChar + hb_DisOutAt 逐像素重繪。使用者認為不適當。

成功：FancyColors 風格 (關鍵！)

完全複製 TABLE/WINFORM 的 FancyColors 模式：

- 建立完整 8 色位 colorSpec (不是簡單 padding)
- 每欄設 defColor := {3, 4} (不是預設 {1, 2})
- 每欄設 colorBlock := {|| {3, 4}}
- 三段式 colorRect : {2,2} + {8,8} + {2,2}

成功原因：defColor {3,4} 和 colorRect {2,2} 之間有明顯的色位差異 (3→2, 4→2)，TBrowse 能正確辨識並重新上色。

五、色彩主題整合

5.1 架構

所有光棒配色統一由 color_theme.prg 的 LoadBrowseHiColors() 管理。定義在 SYSTEM.CH :

```
#define BROWSE_ROW_COLOR 10 // aColor[10] = 列高亮色
#define BROWSE_CELL_COLOR 11 // aColor[11] = 格高亮色
```

啟動流程：

```
STK.PRG:
aColor := { ..., "N/BG", "W+/R" } // 預設值
LoadBrowseHiColors()           // 依主題覆蓋
```

5.2 使用方式

dbEditEx (dbedit_mouse.prg)

```
cC1 := hb_TokenGet(oBrowse:colorSpec, 1, ",")
cC2 := hb_TokenGet(oBrowse:colorSpec, 2, ",")
oBrowse:colorSpec := cC1 + "," +
MEMVAR->aColor[BROWSE_ROW_COLOR] + "," + // pos2=row
cC1 + "," + cC2 + "," +
cC1 + "," + cC1 + "," + cC1 + "," +
MEMVAR->aColor[BROWSE_CELL_COLOR] // pos8=cell
```

TABLE/FancyColors (WINFORM.PRG)

```
oBrowse:colorSpec := "N/" + cBg + "," +
MEMVAR->aColor[BROWSE_ROW_COLOR] + "," + // pos2=row
"W+/" + cBg + ",W+/BG," +
"W+/" + cBg + ",R/RB,R/" + cBg + "," +
MEMVAR->aColor[BROWSE_CELL_COLOR] // pos8=cell
```

5.3 各主題配色表

主題	底色	row 列高亮	cell 格高亮
0, 1	黑	N/BG 青	W+/R 紅

2	黑	N/BG 青	W+/RB 紫
3, 12	棕	W+/B 藍	W+/R 紅
4, 8	黑	N/BG 青	W+/R 紅
5, 11	紫	N/BG 青	W+/R 紅
6, 7	青	W+/R 紅	N/W 白
9	綠	N/BG 青	W+/R 紅
10	藍	N/BG 青	W+/R 紅

5.4 配色原則

- pos2 (row) 和 pos8 (cell) 必須有明顯色差
- pos3/4 (平時底色) 和 pos2 (row) 也要有色差
- 三者互不相同，才能清楚區分「一般行 / 選中行 / 當前格」
- 主控台只有 8 種背景色 (+ modifier 無效)，避免 GR+/GR 等無效組合

5.5 如何修改配色

只需修改 color_theme.prg 的 LoadBrowseHiColors() 函式：

```
/* 例：主題 6 改成綠底 row + 黃底 cell */
CASE nTheme == 6
    MEMVAR->aColor[BROWSE_ROW_COLOR] := "W+/G"
    MEMVAR->aColor[BROWSE_CELL_COLOR] := "N/GR+"
ENDCASE
```

修改後重新啟動程式即可生效。不需改 dbEditEx、TABLE 或任何模組程式碼。

六、總結

6.1 解決的問題

問題	解法
DBEDIT 無滑鼠支援	dbEditEx 取代 · 內建滑鼠事件 SWITCH
hover 變色	MsBrowseKey() 統一處理 K_MOUSEMOVE
PgDn/PgUp 只移一行	down()->pageDown(), up()->pageUp()
光棒只高亮 cell	FancyColors + 三段式 colorRect
光棒配色跟主題	BROWSE_ROW/CELL_COLOR in color_theme.prg
分隔符中斷光棒	defColor{3,4}+colorBlock 讓 colorRect 完整覆蓋

6.2 檔案修改清單

檔案	修改內容
dbedit_mouse.prg	PgDn/PgUp 修正 + 8色位colorSpec + defColor + 三段式colorRect
WINFORM.PRG	FancyColors pos2/pos8 改用 aColor[BROWSE_ROW/CELL_COLOR]
TABLE.PRG	cell colorRect {3,4} → {8,8}
color_theme.prg	新增 LoadBrowseHiColors() 各主題配色
SYSTEM.CH	新增 BROWSE_ROW_COLOR(10), BROWSE_CELL_COLOR(11)
STK.PRG	aColor 擴充到 11 元素 + 呼叫 LoadBrowseHiColors()

6.3 未來擴展

- BROWSE 所有顏色 (底色、標題、負數等) 統一到 color_theme.prg
- DBFWIN/WINDOW BROWSE 光棒改為全列含分隔符 (需進一步研究)
- 其他 browse 模組 (P37, RPTWIN, DAYWIN 等) 套用相同配色