ErrorSys 整合 + 滑鼠 Cell Click 改造

UsrFontErrorHandler 併入 DefError / DEBUG_LOG 等級 /
Alert→MessageBox
ArrayBrowse cell click / TBrowse refreshAll / Table cell color

撰寫日期：2026-05-17

專案：HBSTK

一、ErrorSys 整合

原本兩條錯誤處理路線分散：UsrFontInit 在啟動時安裝 UsrFontErrorHandler 攔截所有錯誤、記詳細日誌到 Debug_Type.txt、然後鏈式呼叫 DefError 處理彈窗。兩條路線維護負擔大、且 ErrorBlock 替換鏈讓 BEGIN SEQUENCE / RECOVER 必須用 ErrorBlock({|e|BREAK(e)}) 暫存的陷阱。

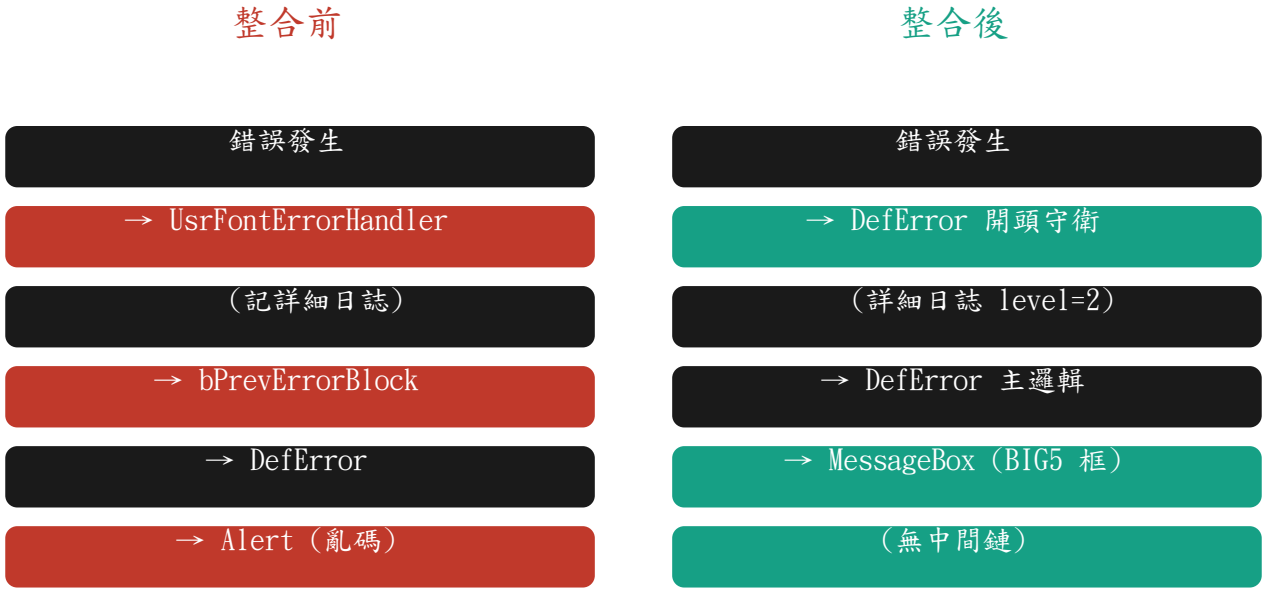

圖 1 ErrorSys 整合前後對照

DEBUG_LOG 等級接起來

stk.ini 的 'DEBUG_LOG=0/1/2' 原本沒有任何程式讀取（死設定）。整合時 WriteDebugLog 新增 'nLevel' 參數（預設 1），讀 stk.ini DEBUG_LOG cache 起來，0 或 nLevel > 設定值跳過。

```
FUNCTION WriteDebugLog( cMessage, nLevel )
  STATIC s_nDebugLog := NIL
  hb_default( @nLevel, 1 )
  IF s_nDebugLog == NIL
    s_nDebugLog := Val( GetIniValue("DEBUG_LOG") )
  ENDIF
  IF s_nDebugLog == 0 .OR. nLevel > s_nDebugLog
    RETURN NIL
  ENDIF
```

Alert → MessageBox

DefError 原本用 ‘Alert(cMessage, aOptions)‘ Clipper 內建，框線在 BIG5 環境下用 IBM-DOS 區段 0xB0-0xDF 被誤判，產生亂碼。改用 STK 自己的 ‘MessageBox(...)‘ 使用 MEMVAR->aFrame BIG5 框線字元。若 aFrame 為 NIL（極早期啟動錯誤）fallback 回 Alert。

四個錯誤檔職責

檔名	寫入者	模式	用途
ErrorMsg.TXT	DefError	ADDITIVE	給客服看的歷史
Error.TXT	DefError	覆寫	最後一次錯誤
Debug_Type.txt	WriteDebugLog	ADDITIVE	開發除錯（含 stack）
error.log	FiveWin (PrnAsk)	覆寫	PrnAsk 的，與 Stk32 無關

二、ArrayBrowse 滑鼠 Cell Click 三合一修補

ArrayBrowse 的滑鼠點 record 行為有三個 bug，使用者報告：「點 cell 沒辦法切換 column」、「點 hint 後游標跑位」。修補三件事：

編號	Bug	修補
A	row 公式 off-by-1（line 336）	nDelta := (MRow()-nTop-nHdrH) - (rowPos - 1)
B	無 column click 邏輯	從 TABLE.PRG 移植 column walk + getColumn:width 累加
C	MsHitTest flash 沒還原游標	HitTest 4 處 flash 後加 SetPos(nSavRow, nSavCol)

Bug A 推理（為什麼錯一格）

MRow() 在 nTop+nHdrH 時 = 點到資料第 1 列（rowPos=1 應該不動）。
舊公式：0 - rowPos = -1 → 錯誤地往上 1 格
新公式：0 - (rowPos - 1) = 0 → 正確不動

Bug C 推理（為什麼游標跑位）

ArrayBrowse 主迴圈 line 174 `nRowPos := Row()` 抓當前游標位置；line 189 `SetPos(nRowPos, nColPos)` 還原。

點底部 [Enter:修改] 提示時：MsHitTest flash 動畫 SAY 到提示位置 → **游標停留在 hint 區（底部）** → 下次迭代 `nRowPos := Row()` 抓到底部 → EditFit 用 `@ Row(), Col() Get ...` → GET 出現在底部。

修補：HitTest 開頭 `LOCAL nSavRow := Row(), nSavCol := Col()`，4 處 flash 後加 `SetPos(nSavRow, nSavCol)` 還原。

三、TABLE.PRG 兩個修補

3-1. K_UP/K_DOWN refreshCurrent → refreshAll

Alt+F3（StockWin）上下移動時，舊列數據殘留為新列內容（BBB 上移到 AAA 後，原 BBB 列變成 AAA）。

根因：Harbour TBrowse 的‘refreshCurrent()‘ 不會自動清舊高亮列，與 Clipper TBrowse 行為不同。

解決：改用‘refreshAll()‘。速度差約 10ms（10-20 列重畫），單次按鍵感覺不到，長按連發可能略卡，可後續加 DispBegin/DispEnd 優化。

驗證：對照 E:\ConvertFW\900\TABLE.PRG（Clipper 原始）確認原始邏輯預期是 refreshAll 行為。

3-2. DoGet 編輯 GET 用 colorSpec index 8

進 cell 編輯時 GET 顏色不是「紅底白字」。原本‘GetNew(..., oBrowse:colorSpec)‘ 取整串，GET 用第一段（normal N/W）→ 黑底白字不是想要的。

oBrowse:colorSpec 8 段索引

1	normal 背景	N/cBgColor
2	row hilite（光棒列）	W+/B 藍底白字
3-7	邊框/footing 等	...
8	cell hilite（選中）	W+/R 紅底白字 ★

圖 2 colorSpec 索引對照

```
cHi := AllTrim( hb_tokenGet(oBrowse:colorSpec, 8, ",") )
IF Empty(cHi) ; cHi := "W+/R" ; ENDIF
oGet := GetNew( Row(), Col(), {|x|...}, "mGetVar",, cHi + "," + cHi )
```

四、踩坑紀錄

坑	症狀	解法
Stk32 還在跑 rebuild link 失敗	BUILD OK 訊息誤導，使用者測舊 EXE 以為 patch 沒生效	rebuild 前先 Stop-Process -Name Stk32 -Force
MsKeyFromLabel 漏 KY_ALT+KY_N	[Alt+N:英轉中] 點不到	DO CASE 補 KY_ALT+KY_N → K_ALT_N
MsKeyFromLabel 漏 CTRL+方向鍵	[Ctrl↓:進入] 等點不到	DO CASE 補 4 對 + CTRL2 版本
colorSpec index 用錯	原本用 index 2（row hilite W+/B 藍底白字）	改 index 8（cell hilite W+/R 紅底白字）
TBrowse refreshCurrent Harbour vs Clipper 不同	Harbour 不會自動清舊高亮列	K_UP/K_DOWN 改 refreshAll
aTitle [...] 框框 + MsKeyFromLabel	prefix 含 [不匹配	傳入前 strip 頭尾 [ ]
MsHitTest flash 殘留游標	點 hint 後游標跑到 hint 位置	HitTest 4 處 flash 後 SetPos 還原

五、影響範圍

本次修補主要影響三個 browse 引擎及兩個系統檔：

引擎/系統	對應程式 / Browse	使用者場景
ArrayBrowse (ARRAY.PRG)	FitList (LIST.PRG)	F4 啟用 Fit picker
Table (TABLE.PRG)	StockWin/PartsList...	Alt+F3 倉庫視窗、F4 配件選擇
MsHandler (mouse_handler)	全系統 mouse hover/click	所有 PRG 滑鼠動作
ErrorSys (ERRORSYS.PRG)	全系統錯誤處理	任何錯誤彈窗
WriteDebugLog (clipper_compat)	全系統日誌寫入	DEBUG_LOG 設定生效

六、未來可加強

A. BRRAY.PRG 有跟 ArrayBrowse 同樣 3 個 bug（refreshCurrent/row 公式/無 col click），已記在 UNDO_LIST，等使用者反映 ListByKind 異常再修。

B. K_UP/K_DOWN refreshAll 加 DispBegin/DispEnd：若長按連發感到卡頓，可包覆批次繪製降低 paint 次數。

C. 移除 PRINT.PRG ErrorBlock 暫存：ErrorSys 整合後不再需要，可以簡化 BEGIN SEQUENCE 區塊。