

P371/P372 InvForm 參考 WINFORM 全面改造

滑鼠支援 + DispBegin/DispEnd + 列印 BUG 修復

撰寫日期：2026-05-16

專案：HBSTK

一、訴求與現象

P371（銷項發票）/ P372（進項發票）共用 InvForm()，使用者反映：

1. BROWSE 區內無法用滑鼠點選資料列、無法用滾輪捲動
2. BottomMessage 功能提示 hover 變色後無法復原
3. SAY 標頭功能提示也無效
4. TAB 切換顯示模式時畫面一行一行刷，非常緩慢
5. Alt+F9 列印爆 BASE/1081 + BASE/1003 雙重錯誤

二、本輪修改項目（按時間倒序）

時間	修改	對照 WINFORM
20:55	PRINT.PRG xValue 型別保護 + BEGIN SEQUENCE 防護	—
20:43	P37 K_TAB 加 DispBegin/DispEnd	line 345-371
20:36	P37 主迴圈 MsBrowseKey + row click + 滾輪	line 36/56/98-134
20:20	P37 補 MsHoverUpdate + MsHitTest	line 73
17:15	P37 K_INS/K_RETURN ReadModal 包圍 (v2 修正順序)	line 641-687
16:10	WINDOW/DBFWIN/TLIB 標題列加 [Alt+H]	—
15:45	WindowMessage lHelp 分支補 MsAddKey	—

三、主迴圈：InKey → MsBrowseKey

WINFORM 早已升級用 MsBrowseKey() 統一處理 hover/click/滾輪，InvForm 還在用原始 InKey + 手動 MsHoverCheck/MsClick，導致：

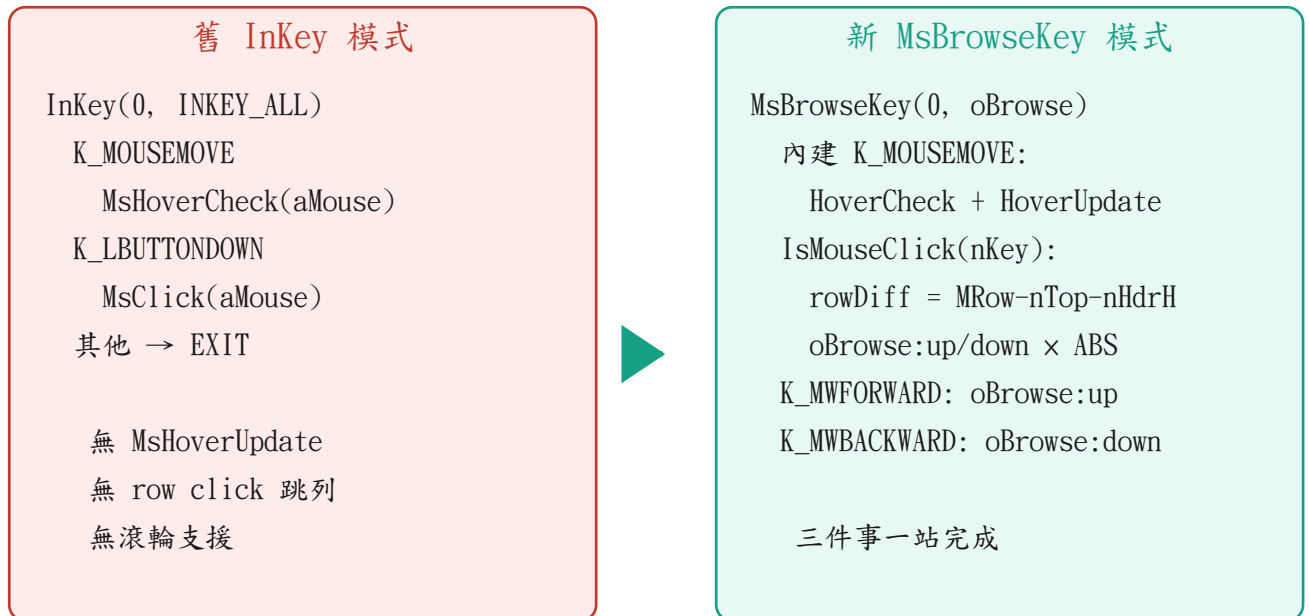


圖 1 主迴圈新舊比較

MsBrowseKey 內部做了什麼

mouse_handler.prg:1195 內部 DO WHILE 迴圈處理：

- K_MOUSEMOVE → HoverCheck(M->aMouse) + HoverUpdate(aMsReg)
- K_RBUTTONDOWN → 回 K_ESC
- K_LDBLCLK 在 browse 區 → 送 K_F10
- K_MWFORWARD/K_MWBACKWARD 在 browse 區 → 回給 caller
- K_LBUTTONDOWN 在 browse 區內 → 回給 caller (row click)
- K_LBUTTONDOWN 在 browse 區外 → MsClick(aMouse) +

MsHitTest(aMsReg)

row click 計算公式

```
nHdrH := IIF(Empty(oBrowse:headSep), IIF(Empty(getColumn(1):heading), 0, 1), 2)
nMsDiff := ( MRow() - oBrowse:nTop - nHdrH ) - ( oBrowse:rowPos - 1 )
IF nMsDiff != 0
  FOR nMsI := 1 TO ABS(nMsDiff)
    IF nMsDiff > 0 ; oBrowse:down() ; ELSE ; oBrowse:up() ; ENDIF
  NEXT
ENDIF
```

四、ReadModal 包圍：MsPushReg 必須在 BottomMessage 之前

v1 把 MsPushReg 放在 BottomMessage 之後，導致 BottomMessage 寫入的 hint zones 立刻被 MsPushReg 清掉。v2 嚴格依 WINFORM 順序：



圖 2 ReadModal 滑鼠包圍 7 步驟

關鍵：MsPushReg 行為

mouse_handler.prg:1104 PushReg() 同時做兩件事：

1. 將當前 aMsReg / aMouse / nHover 壓入 ::aRegStack 堆疊
2. 立即清空 aMsReg / aMouse / nHover

→ 任何 PushReg 之前的 BottomMessage 寫入會被清掉，後續 ReadModal 期間 hover/click 找不到目標。

五、DispBegin/DispEnd 加速畫面繪製

WINFORM 在三個關鍵點用 DispBegin/DispEnd 批次處理畫面寫入：

位置	包覆內容	對應 WINFORM 行
初始顯示	WinOutLine + SHOWBLOCK + InvFormMessage + refreshAll + stabilize	36-56
K_TAB 切換	RestScreen + WinOutLine + display + Message + refreshAll	346-370
WinFormShow 共用	SHOWBLOCK + FOR display	1109-1119

為什麼有效？

GTWVT 每次 SAY/display() 都會 dispatch

WM_PAINT，造成「一格一格刷」。DispBegin 把後續的 SAY 寫入緩衝區，DispEnd 才一次性 invalidate + paint。TAB 切換從「逐行刷」變成「瞬間切」。

六、Alt+F9 列印雙重 BUG 修復

BUG 1: BASE/1081 Argument error: +

PRINT.PR:654 '?? +xValue+' 串接時 xValue 是非字串型別。RptLevel 內部當 cPicture 空 且 nLength 空時，xValue 維持原始 NUMERIC/DATE。

修補 (line 641 後加正規化) :

```
IF ValType(xValue) != "C"
  DO CASE
    CASE ValType(xValue) == "N" ; xValue := AllTrim(Str(xValue))
    CASE ValType(xValue) == "D" ; xValue := DToC(xValue)
    CASE ValType(xValue) == "L" ; xValue := IIF(xValue, ".T.", ".F.")
    OTHERWISE ; xValue := ""
  ENDCASE
ENDIF
```

BUG 2: BASE/1003 Variable does not exist: NINCAMT

PRINT.PR:608 macro 解析時報表 SubForm 引用不存在的欄位 (如 NINCAMT)。用 BEGIN SEQUENCE / RECOVER 防護，配合 ErrorBlock 暫存替換：

```
bSavedErr := ErrorBlock( {|e| BREAK(e) } )
BEGIN SEQUENCE
  ...4 種 macro 取值...
RECOVER
  xValue := ""
  WriteDebugLog( "RPTLEVEL: " + cFrmAlias + " Level=" + cLevel +
    " cField=[" + cField + "] cAliasXXX=[" + cAliasXXX + "] 找不到此欄位" )
END SEQUENCE
ErrorBlock( bSavedErr )
```

關鍵踩坑

BEGIN SEQUENCE 單獨用無效。UsrFontInit 設了全域 ErrorBlock 攔截所有錯誤，必須先 LOCAL 'bSavedErr := ErrorBlock(|e| BREAK(e))' 暫存替換為 BREAK 版本，RECOVER 才會被觸發。END SEQUENCE 後立刻 'ErrorBlock(bSavedErr)' 還原。

七、踩坑紀錄（值得記入 Memory）

坑	症狀	解法
MsPushReg 順序	BottomMessage hint zone 在 ReadModal 看不到	MsPushReg 必須在 BottomMessage 之前
缺 MsHoverUpdate	BottomMessage hover 變色不還原	MsHoverCheck 後補 MsHoverUpdate (aMsReg 區)
STATIC FUNCTION 跨檔不可見	GetColWidth 鏈接失敗	在使用檔內複製一份 STATIC FUNCTION
\x1a (Ctrl+Z DOS EOF)	Harbour 視為檔案結束，後面函式不解析	插入新函式必須在 \x1a 之前
UsrFontInit 全域 ErrorBlock	BEGIN SEQUENCE / RECOVER 抓不到	先 ErrorBlock({ e BREAK(e)}) 暫存替換
xValue 型別漂移	?? + xValue 串接拋 BASE/1081	IF/ELSE 結束後 ValType 正規化

八、未來可加強

A. SAY 標頭點擊跳欄位：目前只有 BUYSELL/QUOT/STOCK 在主體做 SayReg() 註冊。P371/P372 共用 InvForm 沒做。需在 InvForm 開頭或在 SHOWBLOCK 內針對 CusInvM/SupInvM 每個 GET 對應的標籤呼叫 SayReg()。

B. ApplyKey 各 nKey case 內未包
DispBegin/DispEnd: K_UP/K_DOWN/K_PGUP/K_PGDN/K_CTRL_PGUP 等多個 case 都有 'Eval(SHOWBLOCK) + FOR display' 模式，可包裝成 InvFormShow() helper（仿 WINFORM 的 WinFormShow line 1109）。

C. 修報表定義：Debug_Type.txt 會記下哪些 SubForm 行引用了不存在的欄位，依此修 SubForm DBF 把 NINCAMT 等無效欄位改成正確名稱。

D. 其他 PXX 共用 InvForm
模式的程式：P31/P32/P33/P34/P48/... 等。本次改造完整套用於 InvForm，所有共用都自動受惠。