

DIALOG.PRG CodeToName 修正

日期：2026-05-27 | 範圍：P113 零件代碼→名稱轉換失效

一、訴求與現象

P113 零件號碼新增 / 修改畫面中，「車種 / 品名 / 品牌」欄位有特殊機制：使用者輸入代碼後按 Enter，畫面應顯示對應的中文「名稱」（從 Type / ChName / Brand DBF 查表）。

觸發機制（P113.PRG ‘AddParts()’ 中）：

```
AAdd(aElement, { {.F., ""}, {.T., "Type"}, {.T., "ChName"}, {.T., "Brand"}, ... })

■ aElement[9,nIndex] = {lCodeEffect, cPhaseAlias}
lCodeEffect=.T. → ■■■■■■ CodeToName(cPhaseAlias, code) ■■■■
```

症狀：Clipper 29 年正常，Harbour 移植後車種代碼按 Enter 後仍顯示代碼，沒查到名稱。

二、四個 fork 程式碼對照

版本	路徑	CASE 4 結尾「repaint」3 行
Clipper 原版	E:\ConvertFW\900\DIALOG.PRG	沒有
ClassStk 中間版	E:\ConvertFW\ClassStk\source\DIALOG.PRG	沒有
FWSTK FiveWin 版	E:\ConvertFW\FWSTK\source\prg\DIALOG.PRG	沒有
HBSTK Harbour 版（你目前的）	E:\ConvertFW\HBSTK\source\DIALOG.PRG	多加 3 行

三、HBSTK 多加的 3 行（line 696~698）

```
/* repaint field with normal color after InputData (GET leaves N/BG residue)
*/
cString := AnyTypeToString( aElement[3,nIndex] )
```

```
@ nInitialRow, nInitialCol+nTitleLen SAY PadR(cString, nGetLength) COLOR ...
```

註解原意：GET 系統執行 InputData 後可能殘留

N/BG（深底亮綠）顏色，需要 repaint 清除。

問題：repaint 用的是 `aElement[3,nIndex]`（原始代碼），沒有透過 CodeToName 轉換。這 3 行 永遠執行，會把上方 IF Eval 內已正確顯示的「名稱」覆蓋成「代碼」。

四、Harbour 版執行流程（含 bug 點）

```
CASE nSysPara = 4
... InputData() ██████████ ...
IF (K_UP .OR. K_DOWN .OR. K_ENTER)
    IF Eval( aElement[7,nIndex], nIndex )
        cString := CodeToName(cPhaseAlias, code) // ████
        @ ... SAY PadR(cString, ...) // ████████
    ENDIF
ELSE
    ...
ENDIF
/* HBSTK ████ 3 █ (line 696-698) */
cString := AnyTypeToString( aElement[3,nIndex] ) // █ raw code, █
CodeToName!
@ ... SAY PadR(cString, ...) // ██████████
ENDCASE
```

五、解法（先註解，保留以便日後參考）

把 3 行用 // 註解掉，加上 BIG5 中文說明（還原 Clipper 行為）：

```
// === ████████ 3 █ (20260527): GET ██████████, ██████ CodeToName ████ ===
// █ Clipper ████ 3 █ (E:/ConvertFW/900/DIALOG.PRG line 488 ██████ ENDCASE)
// ██████: ██████████, ██████████, ████ CodeToName ████
// /* repaint field with normal color after InputData (GET leaves N/BG
// residue) */
// cString := AnyTypeToString( aElement[3,nIndex] )
// @ nInitialRow, ... SAY PadR(cString, ...) COLOR ...
```

六、踩坑與調查過程

- 找不到 UpdateList / PDF 紀錄：搜尋 HBSTK/doc 內所有 .md 與 .py，沒找到任何提到 GetGeneralInput 或 N/BG residue 的紀錄。推測是當初加這 3 行時沒寫 UpdateList。
- 改用「四個 fork 對照」破解：查 900/, ClassStk/, FWSTK/ 三個版本都沒有這 3 行，只有 HBSTK 有。確認是 HBSTK 獨有的修補。
- 時間軸鎖定引入時間：

時間	DIALOG.PRG 狀態
2026-04-08 ~ 04-12 (log 備份)	沒有 這 3 行
2026-05-26 17:30 (bak_quickbox)	已加入 這 3 行

期間	4/12 ~ 5/26 之間有人加入，未留 UpdateList / 註解只說 N/BG residue
----	---

七、潛在副作用（待觀察）

註解掉 3 行後，原本要解決的「GET 殘留 N/BG 底色」可能重現。可能症狀：
某些對話框欄位輸入後底色變黑底亮綠（N/BG），未自動還原成正常色。

備案：方案 B—保留 repaint 但 cString 也走 CodeToName：

```
// ■■ (20260527): ■■ CodeToName, ■■■■■■■■■■■■■■■■■■■■■■  
cString := IIF( lCodeEffect, CodeToName(cPhaseAlias,  
PadR(aElement[3,nIndex],8), .F.), aElement[3,nIndex] )  
cString := IIF( Empty(cString), aElement[3,nIndex], cString )  
cString := AnyTypeToString( cString )  
@ nInitialRow, ... SAY PadR(cString, nGetLength) COLOR ...
```

若副作用出現，改用方案 B 雙保險。

八、順帶發現未修的關聯問題

同函式 line 691 條件邏輯有誤：

```
IF ( M->nKeyStroke # K_ESC .OR. M->nKeyStroke # K_F12 ) .AND.  
!IsData(M->nKeyStroke)
```

由於 $K_ESC \neq K_F12$ ，nKeyStroke 不可能同時等於兩者，OR 條件永遠 .T.。正確應該用 .AND.：「nKeyStroke # K_ESC .AND. nKeyStroke # K_F12」。

影響：不影響本次 CodeToName 修正，但表示原本想加 K_F12 跳出處理沒生效。列為後續事項，等本次修正穩定後再處理。

九、修改檔案清單

檔案	改動
HBSTK\source\DIALOG.PRG	CASE 4 結尾 3 行 SAY/cString 註解掉 + BIG5 中文說明
HBSTK\source\DIALOG.PRG.bak_repaint_comment_20260527_1634	備份原版

十、Debug 心得（Memory 候選）

1. 找不到文件時的對照策略：不要只看當前版本，要對照所有 fork（Clipper / 中間版 / 其他平台版）。這次能立刻定位元兇就是因為三個 fork 都沒這 3 行 →

確認是 HBSTK 獨有修補。

2. 時間軸鎖定法：看 log/source_backup 等老備份，比較 mtime + 內容，鎖定問題引入的時間區間。

3. 保守原則：未知原因的修補不要盲目刪除，先註解：雖然知道這 3 行造成 bug，但不知道當初加的具體原因。先註解（不刪除）能：

(a) 還原 Clipper 行為驗證主要 bug 修好

(b) 萬一原本的「N/BG 殘留」副作用出現，能快速還原

4. CLAUDE.md 黃金法則再次驗證：「Clipper 邏輯一樣 → Harbour 應該也一樣」。這次同樣是用這個原則破解。