

DBFCDX/1020 DACCDATE 刪除封存漏防呆修正

日期:2026-07-25 | 客戶:虹奇實業(0115)S005 | 檔案:BUYSELL.PRГ / QUOT.PRГ(SaveToDelDbf)

一、訴求與現象

客戶虹奇實業 S005 於 2026-07-25 12:30
回報當機:在日報視窗(EditDayActWin)刪除訂單項目時,系統呼叫 SaveToDelDbf 把原單寫入「刪除封存檔」DelMain,其中結帳日欄 dAccDate 被塞入 NIL(來自 aHead[ACCDATE])→ DBFCDX/1020 Data type error:DACCDATE(日期欄拒收非日期值)。
堆疊:SAVETODELDBF(9853) <- BUYSELL(1500) <- EDITDAYACTWIN <- DAYWIN;ARGS 顯示 [1]U:NIL(第一引數為 NIL)。

二、踩坑與陷阱

項目	內容 / 教訓
靜態追不到 NIL 源頭	BUYSELL:139 無條件 aHead[ACCDATE]:=Date() 初始化,理論上不該 NIL;但 560/1129/1807 三處日期 guard 的「存在」本身證明實務上真的會 NIL。
原開發者也沒消源頭	2026-06-04 註解已點名疑犯路徑(GetOrderByNo F9查單 / PrePrice1Box / EditDayActWin),但當初也是「加 log + guard」處理、未根除;可見 NIL 源頭是資料相依的互動邊界。
07-01 修正漏了這條路徑	2026-07-01 的日期 guard 只補在「寫主檔」三處(560/1129/1807),而「刪除→存封存檔 SaveToDelDbf」(BUYSELL:1500)這條沒補到 -> 同一種 NIL 換入口又炸。
刪除迴圈只重設 ACTDATE	CASE 3 刪除(BUYSELL:1459)只 aHead[ACTDATE]:=(cSubAlias)->dActDate,沒碰 ACCDATE,故 ACCDATE 完全仰賴先前載入是否為有效日期。

現有 log 撈不到本次	既有 log(BUYSELL:1805)在「修改」路徑非刪除路徑;且 log 寫客戶本機非 FTP(錯誤包只上傳 bmp+txt);加上客戶 DEBUG_LOG 未開 -> 本次無任何 log 可查。
--------------	--

三、解決方法(A+B)

A(對症止血):在 BUYSELL:1500 呼叫 SaveToDelDbf 前,補上與 07-01 相同的日期防呆 —— aHead 的 ACC/ACTDATE

若非日期(NIL),改用該筆訂單主檔的真實值

(cMainAlias)->dAccDate/dActDate。刪除訂單不再當機。

B(抓源頭):guard 觸發時 WriteDebugLog 記下 NIL 當下的上下文(單號 / 訂單類型 / 主檔 dAccDate),補上刪除路徑之前完全沒有的

log,供下次重現時回推 NIL 從哪條路徑進來。

```
IF ValType(aHead[ACCDATE]) != "D" .OR. ValType(aHead[ACTDATE]) != "D"
    WriteDebugLog("SAVETODEL-GUARD ACct="+ValType(aHead[ACCDATE])+" ACTt="+...+"
No="+aHead[ORDERNO]+...)
ENDIF
IF ValType(aHead[ACCDATE]) != "D" ; aHead[ACCDATE] := (cMainAlias)->dAccDate ;
ENDIF
IF ValType(aHead[ACTDATE]) != "D" ; aHead[ACTDATE] := (cMainAlias)->dActDate ;
ENDIF
SaveToDelDbf( cUserName, aHead, aBody, ... ) // BUYSELL:1500
```

四、使用技術與用法

技術	用途
FTP 錯誤解析(ErrWatch)	從 .trigger/.txt 抓錯誤碼/堆疊/op/ARGS,定位 DBFCDX/1020 + DACCDATE + NIL
堆疊逐層追蹤	SaveToDelDbf -> BuySell:1500 -> EditDayActWin,定位刪除封存路徑
grep 既有 fix 註解 / log	找到 06-04 診斷註解(1803)、07-01 guard(560/1129/1807)、既有 log(1805),站在前人肩上
BIG5 bytes-only 改碼	PRG 中文註解以 Python bytes-only 寫入 BIG5,避免 UTF-8 混入亂碼

guard + WriteDebugLog 診斷法	對症擋 NIL(不當機)同時記錄異常上下文,下次 重現才能根治源頭
DEBUG_LOG + Commander 抓 log	客戶端 System.ini 開 DEBUG_LOG=1 -> log 寫 W:\S005\Debug_Stk32.txt -> Commander 抓回分析

後續(根治源頭):客戶已開 DEBUG_LOG;待新
stk32.exe(A+B)派版後,下次同情境 guard 會記 SAVETODEL-GUARD
一行,抓回即可回推 NIL 真正來源(疑為 GetOrderByNo)。純 Stk32
端,客戶端需派版生效。