印表機位置/資訊查詢 C 函式化（移除 Python 依賴）

日期：2026/06/03 模組：PrnAsk（fw_prnout.prg、printer_c.c）

一、訴求與現象

列印對話框的「印表機位置」欄一直空白（GDI 與 WebView2 都是）。原本用 get_printer_loc.py（pythonw）查位置，但客戶端根本沒有安裝 Python，pythonw 跑不起來、永遠回空。WebView2 那套 FwGetPrnInfo 改用 PowerShell + wscript，也因為 .ps1 被擋而資訊空白。

二、踩坑與陷阱（繞了一大圈）

#	坑	說明
1	以為位置欄空是 bug	其實當時印表機 Location 屬性本來就空，但應 fallback 到 PortName
2	同步死等 3 秒卡 UI	FwGetPrnLocation 用 FOR 1 TO 30 + idleSleep 死等 pythonw（最多 3 秒），完全沒用 thread
3	改背景 thread + idle 回填	背景 thread 不能碰 GUI；改主 thread idle 輪詢 → 但 FiveWin modal 對話框不觸發 Harbour idle，沒回填
4	改 FiveWin TTimer	TDialog 沒 bTimer，但 TTimer 物件（SetTimer/WM_TIMER）在 modal 下可觸發 → 改用它輪詢
5	空值被永久 cache	原邏輯「連空值也快取」→ 某次查空就永久回空、不再重查。改成「空值不算、會重查」
6	★真正根因：客戶沒 Python	背景查 3 秒仍回空 + W:\S001 沒 get_printer_loc.py → 客戶端無 Python，pythonw 方案徹底不通
7	舊註解誤判 already has	printer_c.c 有舊註解 STK_PRINTERLOCATION，patch 用 in 偵測誤判已存在 → 改用精確 HB_FUNC(STK_PRINTERLOC) 比對

#	坑	說明
8	WebView2 .ps1 被 ExecutionPolicy 擋	FwGetPrnInfo 用 powershell -File prni.ps1，客戶端 Restricted policy 擋掉 → 資訊全空。get_printer_loc 用 -Command 不受 policy，但這裡用 -File

三、解決方法（C 函式，零外部依賴）

STK_PRINTERLOC（C 函式，printer_c.c）：用 Win32 OpenPrinterW + GetPrinterW（PRINTER_INFO_2）直接取 pLocation（有就用）否則 pPortName，BIG5(950) 編碼回傳。純 Win32 API、毫秒級、零外部依賴（不需 Python / pythonw / PowerShell）。

GDI：FwGetPrnLocation 簡化成 RETURN STK_PRINTERLOC(cPrinter)，同步毫秒回 → 整套 pythonw + 背景 thread + TTimer + location cache 全部拿掉，對話框秒開、位置即時顯示。

WebView2：FwGetPrnInfo 的 location 也改用 STK_PRINTERLOC（保證有值）；其餘紙張/邊界查詢的 PowerShell 加 -ExecutionPolicy Bypass 繞過客戶端 Restricted policy。

四、使用技術

技術	重點
GetPrinterW PRINTER_INFO_2	pLocation（印表機內容的「位置」）、pPortName（埠，如 192.168.0.200_lp1）一個 API 全拿；OpenPrinterW 第三參數給 NULL 用預設存取權（唯讀不需管理員）
WideCharToMultiByte(950)	wide → BIG5；輸入印表機名先 MultiByteToWideChar(950) 轉 wide 給 OpenPrinterW
-ExecutionPolicy Bypass	PowerShell -File 受 ExecutionPolicy；加 Bypass 繞過客戶端預設 Restricted
hb_retclen	C 回傳指定長度字串給 Harbour（含中文位置名也安全）

五、成果與教訓

GDI 與 WebView2
列印對話框的印表機位置/資訊都正常顯示，且客戶端零外部依賴（不需 Python、不需 .py 檔部署、不受 PowerShell policy 影響）。

最大教訓：會依賴外部執行環境（Python / PowerShell -File）的功能，在客戶端常常因「沒裝該環境」或「ExecutionPolicy / 防毒」而失效。能用 Win32 C API 直接做的（如印表機資訊），就用 C 函式綁進 EXE，最自包含、最快、最可靠。