

列印對話框取值優化 (持久化 cache + 背景刷新)

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

一、訴求與現象

客戶反映列印對話框「出現很慢」，而且「離開預覽、重新進入再開列印，又要等很久」。對話框在 DEFINE DIALOG 之前同步取 3 樣值：印表機清單 (EnumPrinters)、紙張清單 (DeviceCapabilities)、預設紙張，這些 Win32 列舉在網路印表機環境特別慢，卡住 UI。

二、踩坑與陷阱

#	坑	說明
1	STATIC cache 跨 process 失效	原本用 STATIC hash 快取，但 PrnAsk 每次列印是 Stk32 ShellExecute 啟動的全新 process，STATIC 一定是空的 → 每次第一次開對話框都重查 → 「離開重進又慢」
2	module-level STATIC 宣告位置	把 STATIC s_hPrnCache 宣告放在檔案中間（前面已有函式）→ 編譯錯 E0004「STATIC declaration follows executable statement」。要移到檔案頂部 STATIC 區

三、解決方法 (持久化 cache + 背景刷新)

Phase 1：持久化 cache (跨 process 秒開)

新增 cache 檔 %TEMP%\prnask_prncache.dat，用 hb_Serialize 存 { time, printers, papers, defpapers }。FwCachedPrinterList/PaperList/DefPaper 改成：process 啟動先讀 cache 檔填 STATIC → 對話框秒開；無檔/過期（非當天）才查 Win32 + 存檔。

Phase 2：背景刷新 (thread 安全)

對話框開啟時 hb_threadStart 啟動背景 thread，只做「查最新清單 (local 變數) → 寫 cache 檔」，完全不碰 FiveWin 控件、不碰 STATIC 變數 → 沒有 GUI thread

風險、沒有並發。當前對話框用舊 cache（秒開），下次 process 讀到背景剛更新的新 cache。

四、使用技術

技術	用法
hb_Serialize / hb_Deserialize	把 hash（含巢狀 hash/陣列）序列化成二進位字串存檔、讀回，跨 process 保存
hb_DirTemp	取本機 %TEMP% 路徑（讀取快、各站獨立），cache 檔放這裡
hb_threadStart / hb_threadDetach	MT build 背景 thread。原則：thread 內只碰 local 變數 + 寫檔，絕不碰 GUI 控件或 STATIC 變數（避免 FiveWin GUI thread crash 與並發）
當天有效	cache 用 DToS(Date()) 當時間戳，非當天視為過期重查；配合背景刷新通常保持新鮮

五、成果

第一次（新 process 無 cache 檔）查一次（同原速度）；之後每個 process 都讀檔秒開，徹底解決「離開預覽重進又慢」。背景刷新讓印表機增減隔一次就反映。

log 驗證：重進時出現 CACHE: HIT file + printers/papers FROM CACHE (fast)，取值只花約 60ms（原本要重查 EnumPrinters/DeviceCapabilities）。