

GDI 列印與系統選單整合

工作報告

STK 庫存系統 -- 下午場

日期：2026-04-22 下午

1. GDI 直送列印

1.1 問題

PDF 路徑 (SumatraPDF) 列印品質差：

- * HScale 壓縮 77%/66% 造成字型變形
- * 筆畫變成點狀，點跟點之間空隙明顯
- * ASCII 用 Courier (瘦長) 而非中文字型 (圓滿)
- * CreateFont 寬度計算錯誤 (/2 不該用於 ASCII)

1.2 解決方案

建立 GDI 直送列印路徑，繞過 PDF/SumatraPDF：

gdi_print.prg (C 函式):

GdiOpenPrinter() 開啟印表機 DC
GdiStartDoc() 開始列印工作
GdiCreateFont() CreateFont(nHeight, nWidth) 獨立設定寬高
GdiTextOut() TextOutW 輸出文字
GdiDrawLine() 畫線 (底線/上線)
GdiDrawBitmap() StretchDIBits (造字 bitmap)
GdiEndDoc() 結束列印

prn_to_pdf.prg:

GdiPrintPrn() 入口函式
GdiFlushText() GDI 版 FlushText
FlushText() s_lGdiMode 自動分流 PDF/GDI

1.3 關鍵技術決策

- * ASCII 用 CJK 字型 (~C 預設：半形字用中文字型，參考 UNICODE.INI)
- * ASCII $nFontW = nCellW * DPI / 72$ (填滿半形格位，不除2)
- * CJK $nFontW = nCellW * 2 * nScale * DPI / 72 / 2$
- * 座標轉換：PDF(pt, bottom-up) -> GDI(device units, top-down)

2. 倚天控制碼修正

對照倚天控制碼規格，修正多項解析問題：

- * TX/TY 預設值 = 24 (不是 0)，匹配規格
- * ~T 設定時同步更新 TX/TY (後面蓋前面原則)
- * GetCellW 直接用 TX，GetFontPt/GetLinePt 直接用 TY
- * ~I 不重置邊界 (ML/MR/MT/MD 是版面控制，不在 ~I 範圍)
- * 預設邊界 = 0 (印表機內定值，非硬編碼 200)
- * nCharAdv 單位 0.4 -> 0.2 (ForX 公式 /2 同時作用於 X)

3. 系統選單與對話框整合

3.1 列印路徑整合

三個列印入口都連接到 GDI：

1. PrnAsk 預覽中的「列印」-> FwDoPrint -> GdiPrintPrn
2. PrnAsk 直接列印 -> FwPrnPrintTo -> GdiPrintPrn
3. STK 直接列印 (SKIPASK=1) -> SpoolDirectPrint -> GdiPrintPrn

3.2 SKIPASK 設定統一

DosPntGo 路徑讀 PNTGO_SKIPASK，但對話框寫 PRN_SKIPASK。

修正：CallDosPntGo 同時檢查兩個設定 (取較大值)。

SKIPASK=1 時直接 GDI 列印，不啟動任何外部程式。

3.3 DosPntGo 對話框修改

- * 視窗抬頭「DosPntGo」->「請選擇印表機」
- * 「Skip next」->「下次不再詢問」(中文)
- * 勾選時透過 FwSaveSkip 寫入 PRN_SKIPASK

3.4 fw_prnout.prg BIG5 亂碼修復

fw_prnout.prg 的 BIG5 中文被 UTF-8 replacement characters (EF BF BD) 破壞。

原因：Edit tool 讀寫時將 BIG5 誤轉為 UTF-8。

修法：從 BIG5 備份 (bak20260416) 還原，GDI 修改用 Python bytes-only 重新套用。

3.5 預覽/直接列印 toggle

選單文字交換 (顯示可切換的動作，不是當前狀態)：

- AUTOFORMAT=1 (直接列印) -> 選單顯示「預覽列印」
- AUTOFORMAT=2 (預覽列印) -> 選單顯示「直接列印」

PRN_AUTOFORMAT=4 時走預覽 (非阻塞 hb_processOpen)，不走 GDI 直接列印。

4. 踩過的坑

[1] s_page==NIL 擋住 GDI FlushText

症狀	GDI 模式沒有 HPDF page FlushText 入口檢查 s_page==NIL -> 直接 return
解法	GDI 分支放在 s_page 檢查之前

[2] ASCII 用 Courier 而非中文字型

症狀	規格預設 ~C (半形字用中文字型) 但程式碼用 Courier -> 英文瘦長
解法	參考 UNICODE.INI 改用 s_cCJKFontName (MingLiU)

[3] ASCII CreateFont 寬度太窄

症狀	$nFontW = nBatchW * DPI / 72 / 2 = 6.5 \text{ dev}$ 應該是 $nCellW * DPI / 72 = 12 \text{ dev}$
解法	ASCII 不除2，填滿半形格位

[4] fw_prnout.prg BIG5 被 Edit tool 破壞

症狀	Edit tool 讀寫 BIG5 檔案時 將 BIG5 lead bytes 替換為 EF BF BD
解法	從 BIG5 備份還原 所有修改用 Python bytes-only

[5] PNTGO_SKIPASK vs PRN_SKIPASK 不一致

症狀	對話框寫 PRN_SKIPASK CallDosPntGo 讀 PNTGO_SKIPASK (不存在)
解法	CallDosPntGo 同時檢查兩個設定

[6] [printer] 佔位符未替換

症狀	SKIPASK=1 跳過對話框時 命令中的 [printer] 沒被替換 -> error 1801
解法	替換 [file.prn] 後也替換 [printer]

[7] hb_run 阻塞 console

症狀	hb_run 等待 PrnAsk 完成 但阻塞了 GTWVT console -> 對話框無法顯示
解法	改回 hb_processOpen (非阻塞)

[8] RunAndWait 阻塞 STK32

症狀	自製 RunAndWait (CreateProcess + WaitForSingleObject) 有 message pump 但 STK32 仍凍結
解法	移除 RunAndWait 改回 hb_processOpen

[9] SetColor("") 造成黑底

症狀	StkTogglePntGoFmt 等函式用 SetColor("") 重置成空值 (黑底) 期間 idle repaint 用錯誤顏色
解法	SaveColor + SetColor(cOldColor) 恢復原色

[10] HookSysMenu 未生效

症狀	hb_gtInfo(7) 沒回傳正確的 GTWVT HWND WM_INITMENUPOPUP hook 從未觸發 CheckMenuFlag 永遠 false
解法	移除整個 hook 機制 (未找到正確取得 HWND 的方法)

5. 剩餘待辦 (UNDO LIST)

1. SetColor 恢復 -- VerticalMenu/PickList/Display() 結束時顏色未恢復
2. SetColor 恢復 -- 系統選單點選後顏色未恢復

兩個根因相同：SetColor 改了沒在所有路徑恢復原色。

6. 修改檔案清單

- * compat/gdi_print.prg -- GDI C 函式 (11個)
- * compat/prn_to_pdf.prg -- GDI 模式分流 + 控制碼修正
- * compat/print_spool.prg -- 列印流程 + 系統選單 + SKIPASK
- * PRNASK/source/fw_prnout.prg -- FwDoPrint/FwPrnPrintTo GDI + BIG5 修復
- * PRNASK/source/dospntask.prg -- DosPntGo 對話框中文化
- * source/main.prg -- SetKey(9997) 新增

7. 結論

[OK] GDI 直送列印完成，字型品質與 DosPntGo 一致

下午從字型品質問題開始，建立了完整的 GDI 直送列印路徑，
並整合到 STK 的三個列印入口 (PrnAsk 預覽、PrnAsk 直接、STK 直接)。

同時修正了倚天控制碼解析 (TX/TY/邊界/~I)、
DosPntGo 對話框中文化、SKIPASK 設定統一、BIG5 亂碼修復等。

過程中踩了 10 個坑，包括 PDF font size 限制、
BIG5 編碼破壞、跨 process 通訊困難、console 阻塞等。

最終採用「預覽走 PDF、列印走 GDI」的雙路徑設計，
在保持預覽功能的同時達到與 DosPntGo 相同的列印品質。