

GDI

列印字寬校準(黑體等任何字型強制塞進字元格)

一、訴求與現象

客戶把報表字型(System.ini [Font] CJK_DEFAULT)改成微軟正黑體後,估價單變黑體且單價/數量/小計/備註全部漂出自己的格子;改回細明體就正常。

目標行為=DOSPNTGO:不管什麼字,一律照字元格走(ASCII半格、中文全格),英文中文永遠整齊。

二、踩坑與陷阱

坑	說明
lfWidth 是「平均字寬」	CreateFont 的 lfWidth 對等寬字型=每字精確寬;對比例字型只是平均值,每個字形實際寬各自不同——細明體準、黑體全面失控
實測數據	150dpi 格寬 12px:黑體全形字 26px>目標 24px;ASCII 的 W 寬 26px=格寬 2.2 倍——欄位「跑到別人家」的最大元兇
引擎定位其實沒錯	GdiFlushText 本來就逐字格子定位(每字推進固定格寬);錯的是字形畫太寬,往右溢出視覺蓋到鄰欄——修「字形寬」而非「位置」
等寬 vs 比例的本質	細明體:中文恰 1 全形格/ASCII 恰半格,為報表設計;正黑體:比例字型、side bearing 大,為螢幕閱讀設計,天生不進格子
hbm2 -01 BCC ICE	hbm2 預設最佳化編 gdi_print.c.c 觸發 BCC 內部錯誤(單檔編譯正常);測試 harness 用預編 obj 繞過,正式 hbp 不受影響

三、解決方法

新增 GdiCalibFontW 字寬校準:建字型 →

GDITEXTTEXTENT(GetTextExtentPoint32W)量實際字形寬 → 按比例修正

lfWidth → 微調迴圈保證字形寬 ≤ 目標格寬。

項目	內容
CJK 分支	樣本「測」,目標 2 字元格寬
ASCII 分支	樣本最寬的 W,目標 1 字元格寬;條碼/EF 字型跳過
快取	key=字型 高 寬 目標 粗體,同組合只校準一次(逐字呼叫零成本)

項目	內容
安全性	量測失敗回傳原值=完全維持原行為;細明體校準結果=原值(既有客戶 100% 不變)

驗證:獨立單元測試(真 GDI 記憶體 DC)——細明體 24=24 零改變;黑體 26→24 PASS 塞回格內。

四、使用技術與用法

1. 修改檔案:HBSTK\c_code\gdi_print_c.c(+GDITEXTEXTENT)、HBSTK\compat\prn_to_pdf.prg(+s_hCalibW/GdiCalibFontW+GdiFlushText 兩分支掛勾)。共用檔→Stk32 與 PrnAsk 都要重編派送。
2. 排版鐵律:點陣格式報表=字元格排版;任何字型都必須「量測後塞格」,不可信任 lfWidth 語義。
3. 使用者體感:黑體欄位對齊恢復;黑體 ASCII 被壓進半形格會比原來窄(格式報表必然)。
4. 備份:gdi_print_c.c / prn_to_pdf.prg .bak_calibw_20260722_0046。