

Full Double Buffer v3 完成報告

日期：2026-04-30 專案：HBSTK 狀態：已完成

一、訴求與現象

GTWVT 渲染有兩個獨立的 double buffer (PaintText 和 ScanAndDrawUda 各一個)，造成兩次螢幕寫入。中間有可見的過渡狀態 (ASCII 已畫、UDA 未畫)，可能造成閃爍。

二、根因分析

原始流程有 4 次螢幕操作：

- * 1. PaintText: screen $\rightarrow$ s_hdcPaintBuf (捕捉)
- * 2. PaintText: s_hdcPaintBuf $\rightarrow$ screen (寫 ASCII) $\leftarrow$
中間狀態可見
- * 3. ScanAndDrawUda: screen $\rightarrow$ s_hdcDblBuf (捕捉)
- * 4. ScanAndDrawUda: s_hdcDblBuf $\rightarrow$ screen (寫 UDA)

步驟 2 和 4 之間，使用者可能看到「只有 ASCII、沒有 UDA」的畫面。

三、踩坑與陷阱

- * v1 (WndProc 建新 buffer + 全螢幕 BitBlt) $\rightarrow$ 無法輸入
- * v2 (WndProc 建新 buffer + dirty region BitBlt) $\rightarrow$ 仍然無法輸入
- * v1/v2 失敗原因：WndProc 內的 GetDC/BitBlt 與 PaintText 的 BeginPaint/EndPaint 衝突，可能造成 WM_PAINT 重入或訊息迴圈阻塞
- * v3 成功關鍵：不在 WndProc 做任何螢幕操作，完全由 ScanAndDrawUda 負責唯一的螢幕寫入

四、解決方法 (v3)

最小改動，不建新 buffer，不改 caret，不改 HideCaret：

* WndProc：設 g_bSharedPaint=1 → CallWindowProcW → ScanAndDrawUda → g_bSharedPaint=0（只加 2 行）

* PaintText：g_bSharedPaint=1 時跳過 final BitBlt，匯出 s_hdcPaintBuf 指標到 g_hPaintBuf

* ScanAndDrawUda：g_hPaintBuf

有值時從它捕捉（而非螢幕），然後正常疊加 UDA、正常寫回螢幕

結果：螢幕只有 ScanAndDrawUda 的一次 BitBlt 寫入（含 ASCII + UDA），無中間狀態。

五、修改檔案

檔案	修改內容
usrfont_c.c	新增 g_bSharedPaint (flag) 和 g_hPaintBuf (buffer 指標)
usrfont_wndproc.c	WM_PAINT 中 CallWindowProcW 前後設/清 g_bSharedPaint (2 行)
gtwvt.c PaintText	g_bSharedPaint 時跳過 final BitBlt，匯出 s_hdcPaintBuf
usrfont_scan.c ScanAndDrawUda	g_hPaintBuf 有值時作為 capture 來源（取代 hdc/screen）