H.264 遠端桌面強化報告

STK Commander / Agent — 串流遠端桌面 2026-06-06

一、訴求與現象

原本的遠端螢幕採 JPG 輪詢(一張張截圖傳),在 4K 高解析度站台又慢又卡、操作遲鈍。目標是改用 H.264 幀間壓縮串流,達到省頻寬、流暢、4K 也順的遠端桌面,並可遠端鍵鼠操控。

導入過程實際遇到:H.264 一度「完全不可用」(畫面卡死)、滑鼠位置對不準(尤其 4K 與最大化)、切換畫質會死當、畫面同時出現兩個游標、某些站台完全沒畫面、Relay 掉線等一連串問題,逐一排除後完成。

二、功能強化項目

項目	內容與效果
串流流暢度	改 H.264 串流取代 JPG 輪詢;搭配解析度縮放,4K 站 fps 由 3.7 提升至約 21(流暢 5 倍)
滑鼠精準對位	座標換算改用「實際顯示的畫面尺寸」+ Agent 自動回報遠端真實解析度(支援 4K=3840×2160、多螢幕),最大化、任何縮放都精準
縮放比例可選	抬頭 ComboBox 100%~40%(相對遠端真實解析度);抬頭即時顯示真實解析度與 fps「H.264 W×H fps 操控中」
FPS 可調	15 / 10 / 8 / 5;大解析度配低 fps、小解析度配高 fps 最平衡
遠端游標開/關	可選顯示或隱藏遠端電腦游標(避免本地+遠端兩個游標)
檔案傳輸	「檔案」按鈕開雙面板(本機/遠端 互傳)+ 拖檔進視窗直接上傳遠端 C:\TEMP
視窗操作	加最大化/最小化按鈕、初始視窗加大、可自由縮放
無黑窗	ffmpeg 子程序加 CREATE_NO_WINDOW,開啟/切換不再跳出 CMD 黑窗

三、分發機制(pusher/ffmpeg 自動部署)

H.264 推流需要每台被觀看的站台具備三個檔案:Stk_Agent.exe(常駐)、pusher.exe(推流)、ffmpeg.exe(抓桌面+編碼)。原分發只含 Agent,導致缺 pusher/ffmpeg 的站台「完全沒畫面」。已將三件套納入既有更新鏈:

環節	處理
stk_uploader(收集→FTP)	pack 階段自動抓 pusher.exe(來源 tools/dist)+ ffmpeg.exe(來源 StkAgent/build)一起打包上傳 FTP
stk_updater(FTP→各站)	deploy_socket 部署各站時一併 copy pusher/ffmpeg(file_changed 增量:pusher 變才傳、ffmpeg 不變不傳)
維護重點	每次 rebuild pusher.exe 後置於 tools/dist,uploader 自動取最新;ffmpeg 不常變,放 StkAgent/build 一份即可

四、踩坑與陷阱(問題→根因→解法)

問題	根因	解法
完全不可用/卡死	每 33ms 用 SmoothTransformation 高品質縮放,GUI 執行緒 CPU 爆滿	改 FastTransformation(快數倍)
滑鼠最大化跑掉	用固定尺寸重算縮放,與實際顯示對不上	改用實際顯示的 pixmap 尺寸換算
4K 站滑鼠偏移	寫死 1080p 當基準,4K 桌面全錯	Agent 回報虛擬螢幕真實解析度,動態套用
切換畫質死當	用新尺寸解析舊殘留影格,尺寸不符使 PyQt 直接中止	影格長度檢查跳過 + try/except 防中止
切換後鍵鼠死	connect(最多 8 秒)在 GUI 執行緒,卡住整個介面	串流啟動移到背景執行緒,connect 不凍 GUI
切換後混流變慢	新 pusher 啟動未先關舊的,多個 pusher 同站搶 CPU	Agent 啟動新 pusher 前先 taskkill 舊 pusher
兩個滑鼠游標	gdigrab 預設把遠端游標也抓進畫面	-draw_mouse 參數化,可開關
某站完全沒畫面	該站缺 pusher.exe / ffmpeg.exe	補檔 + 納入自動分發鏈
Relay 掉線不復	watchdog 未運作,Relay 掛了不自動重啟	在 SERVER 本機重新啟動 Relay(待補 watchdog 排程)
同機站台連不上	Agent 連外網 DDNS 經路由器 hairpin 不通	Agent 連線加 127.0.0.1 退路

五、架構與資料流

被觀看站台(Agent→pusher→ffmpeg 抓桌面壓 H.264)→ Relay(9530 依封包標頭分流,純轉發不解析)→ Commander(ffmpeg 解碼顯示);鍵鼠走 Relay 控制通道回送遠端注入。

關鍵:H.264 串流複用 Relay 既有的 9530 控制埠(依連線標頭 role 自動分流 control / 串流),不需在路由器額外開埠,符合「用 Relay 就是為了不動路由器」的原則。

六、使用技術與用法

技術	用途/用法
ffmpeg gdigrab + libx264	pusher 抓桌面(-draw_mouse 開關游標)→ scale 到指定解析度 → libx264 ultrafast+zerolatency 編碼 → mpegts 推流
ffmpeg 低延遲解碼	Commander 端 -probesize 32 -analyzeduration 0 -fflags nobuffer -flags low_delay,首幀延遲 11.8s 降到 2.4s
Relay 標頭分流	連線先送 length-prefixed JSON 標頭,role=pusher/viewer 走串流、type=agent/commander 走控制,同一埠分流
PyQt5 H264ScreenDialog	背景執行緒收流/解碼、generation 標記管理切換、QTimer 顯示最新影格(丟積壓降延遲)
PyInstaller	pusher.exe / Stk_Commander.exe / stk_uploader.exe / stk_updater.exe 打包(32-bit)