

# 为什么会这样？

## ▲最开始,我只是想给文件夹换个色

### >一堆文件夹里只有一个文件夹存放源文件

我想给它赋予一些更好让我区分出它的特性,而文件名前面加上 AAA 太过俗气了.

是的,我开始思考为什么 Windows 不能像 mac 一样改系统的图标(能改,只是要在属性改,而且 Windows 不准备多一些默认的图标给你换,在网络上想找图标也是各种广告而且不符合审美,而美化包对系统的改动太大,不想用).

总而言之我打算自己做一些图标.....

### >Windows 只能使用 ico 文件做图标

时间过了一个钟,我做出了几个 256\*256pixel 的文件夹图片,如下,还是挺好看的嗯!



马卡龙的配色^\_^

但是它是 png 和 svg 格式的,我发现系统只能识别 ico 格式的位图.....



The file E:\Personalfile\Art\paint\2025\10-21shell\_ico\_make\Artboard 3 copy 5.png contains no icons.

OK

(该文件没有包含 icons)

好吧 Windows 你赢了~

png, jpg, ico 都是位图为啥只支持 ico?我也没有心思找这个问题的答案了,我知道我没有那么多时间耗在这里.

## >这个世界没有人关注 icon

我连一份规范的 windows-icon 文档都搜不出来(我冥冥之中感觉肯定有这个东西,作为用户能看得见的界面元素怎么可能没有设计规范)

我从零零散散的信息之中收集到 icon 文件一般分大小约束,分别为 16 24 32 64 128 256 512 1024(pixel),很熟悉对吧,但是为啥有一个 24?在这里就让它继续保持神秘吧!

于是乎我重新导出了一遍 256 大小的图标

## >我只是想转个文件格式 TAT

是我想的太简单了

网上的转换软件都是上古时代的产物(这个世界没有人关注 icon),安装包散落在各种奇怪怪的小网站里,一不小心中了 360 的招:

我都选了本地下载了,一运行结果尼玛的跳出一个 360 下载器下载中差点吓的我心肌梗塞随即我使用超本能反应从任务管理器马上拦截了它,好悬,差点给我电脑安装 360 全家桶.

吃一堑长一智,我选择自己用 ffmpeg 转换,正好我是一名对捣鼓感兴趣的程序员,电脑上之前为了转换音乐格式安装了 ffmpeg 库.

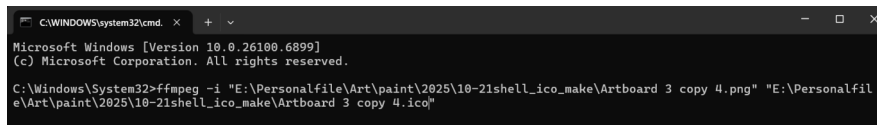
我其实挺好奇 AI 对这种上古时代的库是怎么评价的:

FFmpeg在我眼中简直是数字媒体领域的“活化石级神兽”——它不是上古遗老，而是历经25年迭代依然统治着音视频处理的**基石级存在**。作为程序员，你的选择正印证了那句真理：“FFmpeg可能不是最优解，但永远是最终解”。

省流---25 年前诞生的神.

因为它除了音频编码,图像编码,还可以视频编码,各种格式的文件能相互转换的同时还可以设置各种参数,神中神了好吧~

>为啥只能一个文件一个文件丢进 cmd 输出,还要打字好麻烦好浪费时间啊啊啊啊啊啊



```
C:\WINDOWS\system32\cmd. X + v
Microsoft Windows [Version 10.0.26100.6899]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\System32>ffmpeg -i "E:\Personalfile\Art\paint\2025\10-21shell_ico_make\Artboard 3 copy 4.png" "E:\Personalfile\Art\paint\2025\10-21shell_ico_make\Artboard 3 copy 4.ico"
```

是的,人类的本能占据了我的理智,懒惰充斥着我的大脑,神经突触释放的生物电信号告诉我:"嘿!伙计,你需要一个批处理程序去解决他,这会使你事半功倍的!"

为什么我不直接找一个开源项目,肯定有的吧?

懒得看英文.....不是硬啃不了,而是翻译掉档次.

于是乎我打算自己写一个批处理程序以后也能用他转换别的文件什么的.

>一直到结尾前都很顺利,为什么剧情不是按我想象的走

我在一个特别特别神秘的 bug 上卡了 5 个钟,先说理解吧,我觉得是 Windows 的特性引起的 bug,即使找到了此 bug 的特性并能准确规避,我还是觉得他在我心中如同克苏鲁一样不可名状,让我深深地明白了比特世界的冷漠与恐怖,与知识带来的危险性.....

我先从简单的地方开始入手,就是输出我带入的参数,什么意思呢,就是我选中一堆文件,拖放到 exe 程序图标上后从 cmd 分别输出其父级路径,文件名和文件后缀.

这里要了解 Windows 的一个特性,就是程序运行时会自动传入拖进图标的文件的参数,其中文件数量会以整数形式传入,文件路径会以字符串的形式传入,也就是大家熟知的超链接,就像这样:

```
"E:\Personalfile\Art\paint\2025\10-21shell_ico_make\Artboard 3 copy 4.png"
```

这些都是小菜一碟,只要我能输出文件的路径和改变了后缀的路径,我就可以带入 system()作为 ffmpeg 的运行传参了.哈哈哈哈哈我简直是天才~

## >然而,问题发生了,为什么有概率运行失败闪退?

毫无规律可言!!!!!!不能稳定复现,程序只要调用 ffmpeg 时就会闪退,而且一点报错信息都没有.....

只要传入 3 个以上文件路径,程序的闪退概率高到 90%以上,越多文件越高概率,而且我测试并发现闪退概率和文件数量不呈线性关系.也就是说,这些特性都指向了资源管理和内存访问这一类型的问题.

于是乎我在各种需要给指针分配内存的地方都做了分配检测,不行×

在传递参数的地方做了传参检测,不行×

那就 Dubug 吧.....

卧槽!Debug 时带入 20 个文件都不闪退,能正常输出,不能复刻闪退不行×

## >Ok,高潮的地方来了,他居然和 exe 运行的工作目录有 100%的关联!

工作目录是什么,比如你的 picacomic.exe 放在---"C:\work\study\learn\file\学术研究

资料\ "---这个目录下,他的工作目录就是这里.

我是怎么意识到这几乎没有正常人类会考虑的因素的?

在一开始,我被迷惑了,我有俩个输出程序,都是一样的代码,只是一个输出时 gcc 指令带了-g 参数,用于 debug,它在".\output\debug\"目录下.(就叫他-gexe 吧)

而另一个作为正常输出的程序在".\output\"目录下.(这个就叫 exe)

这两个应用程序明显在不同目录,对吧!

当我在 launch.json 配置好"args": [

```
"E:\\Personalfile\\Art\\paint\\2025\\10-21shell_ico_make\\Artboard 3 copy 2.png",  
"E:\\Personalfile\\Art\\paint\\2025\\10-21shell_ico_make\\Artboard 3 copy 4.png",  
"E:\\Personalfile\\Art\\paint\\2025\\10-21shell_ico_make\\Artboard 3 copy 3.png", ],
```

(这里表示运行时传入这些路径,和拖入的效果一样)

美美开始 debug 找原因(运行的是-gexe),它居然不闪退了!?!当我兴奋地以为-g起作用了,debug 参数设置拯救了一切的时候,现实给我泼了一盆冷水,因为我从外面拖文件进-gexe 也是一样闪退.

## >为什么会这样?

我开始思考这两种启动方式有什么区别:

权限?-----用管理员权限试过了,不对×

库文件?-----我只有一个库文件,没做版本区分和备份,不对×

环境变量?-----我的 VSCode 配置的系统的环境变量,没改没新增,更不对×

最后是工作目录,exe 文件都在同一个目录,不对,吗?

我才知道,exe 的工作目录不是定死在其存放位置的,当我拖入另一个路径的文件到 exe 时,其工作目录会跑到拖入文件的目录下!!!

这有什么区别?不就是运行的工作目录不一样吗,难不成就这个父级目录有问题?

正常人都会这么想吧,可是问题就在这里,我一开始甚至特意创建的纯英文的工作路径,结果不仅中文路径没有问题,除了它之外的所有路径都没有问题!!

这踏马太奇怪了,我完全复制了这个路径下的所有文件,粘贴到上一级,下一级,结果都能正常运行,总之控制变量法都试完了,就是 vscode 的工作目录的父级不行.

> 当一个 bug 能够稳定规避,那意味着问题到这里就解决了,即使它还存在,并没有消失

如果我在别的路径创建一样的工程会不会这样(是不是项目路径的问题),

如果调用除了 ffmpeg 以外别的程序会不会这样(是不是 system()的问题),

如果包含 ffmpeg 的库文件直接调用函数而不是调用 exe(是不是程序调用的问题),

如果重启电脑能不能修复这个 bug(是的我还没有重启过),(是不是.....玄学),

> 好吧,你真有耐心,居然看完了,因为被 Windows 神秘"特性"折磨道心破碎,就写下了这一摊子糟心事

