Frequently Asked Questions
25 February 1996

In this section, you will find a list of frequently asked questions (FAQ) about ThumbsPlus, culled from our electronic mail and telephone support calls. As this section grows, we'll organize it further. If you have questions that you think should be in this section, please submit them to Cerious Software, Inc. Use Internet mail if possible (pcrews@cerious.com), or fax them to 704-529-0497.

What are the limitations when running on Windows 3.1 or 3.11?
- Long file names are not supported.
- The .CUR (Windows cursor) and .ANI (Animated cursor) formats are not supported.
- ThumbsPlus must be able to load a file into a single memory block; the maximum size of these blocks is 16 Mb. The amount of physical RAM in your machine may further limit the size.
- Contact sheets can only be generated at the current display color depth or less. For example, you cannot generate a truecolor contact sheet when using a 256-color display driver.
- The Windows 95 recycle bin is not available.

What is the largest image that ThumbsPlus can load?
Except on Windows 3.1, where the maximum is 16 Mb, there is no fixed limit — the largest file that ThumbsPlus can load on your machine is dependent on the amount of available memory in it. ThumbsPlus does use virtual memory for loading images, so it may help to increase the amount of swap space available.

Be aware that images which are compressed will take much more RAM than disk space; they are decompressed to load and display. In general, you can compute the memory required based on the dimensions and color depth of the image. The *area* of the image times the *bytes per pixel* will give a good estimate of the memory required. For example, a 1000 x 1000 24-bit image would require 1000 x 1000 x 3 = 3,000,000 bytes (24 bits per pixel = 3 bytes per pixel) .

What's the best format for storing grayscale or color photographic images?
If you plan to edit, crop or adjust them, I recommend storing in LZW TIFF format. This format is widely supported and provides good compression. Storing with separate color channels sometimes improves the compression ratio, but is not supported by all applications. For compatibility with applications that do not support LZW TIFF, I recommend PackBits compressed TIFF or Targa.

If your image is "final form," and you do not intend to edit it further, JPEG is a reasonable option for storage. It offers excellent compression, but there is some loss of image accuracy associated with it. I recommend a quality setting of between 70 and 90.

What's the best format for storing drawn image files of 256 colors or less?
GIF or TIFF work well for these. Targa, PCX and BMP can also be used; BMP loads the fastest but is not compressed.

What's the best format for storing bi-level (bi-tonal, monochrome) image files?
I prefer TIFF with CCITT Group 4 compression.

Why do my GIF files look bad when converted to JPEG?
The JPEG format is designed to compress real-world 24-bit color images. GIF files have already been reduced (usually by dithering) to 256-color (8-bit), so most of the original color information is lost. When they are converted to JPEG, the sharp transitions between neighboring pixels are not handled very well. The *Smoothing* option in the *Save JPEG Options* dialog box may improve the situation somewhat.

When I display JPEG or other 24-bit images, they appear grainy.
ThumbsPlus quickly dithers 24-bit images for display on 8-bit displays. This does not show the true color detail

of the original file; you will get much better display results by running a Truecolor (24-bit) or Highcolor (16 bit) display driver, if you can for your display adapter.

Note that you can set JPEG files to load as 8-bit images, using a dithering algorithm in the JPEG library which is better (but a bit slower) than the internal dithering. PhotoCD files may also be dithered to 8-bits while loading. Both of these loading options are available from the Options—>File Types property sheet.

How come my JPEG files seem less clear when I edit them and save them many times?

JPEG format should not be used for "works in progress." Re-saving JPEG files introduces another generation of loss error. For my own stuff, I always save as TIFF (LZW-compressed) during interim steps, and only convert to JPEG for external release (i.e., to a Web page).

Help! I made JPEG files, but my other applications can't read them.

ThumbsPlus provides the option to produce progressive JPEG files, but this format is not yet widely supported. It is useful in specific situations (i.e., Web pages that are going to be accessed only with PJPG-compatible browsers such as Netscape 2.0). The progressive format does not offer any advantage in size for local storage; it is designed for being able to view an image in progressing amounts of detail while being downloaded over slow links.

Why can't I thumbnail some of my CorelDRAW! files?

ThumbsPlus can only read the thumbnail preview directly from CDR, CMX and PAT files. If the files were not saved with a preview bitmap, then ThumbsPlus will report "error reading file." Note that you *can* set up ThumbsPlus to use OLE with CorelDRAW! to thumbnail and view the files, but this is very much slower than reading the preview bitmap directly.

Why won't ThumbsPlus read some of my Photoshop 3.0 files?

ThumbsPlus does not parse the entire Photoshop 3.0 file format; instead, it uses the image from the last view in Photoshop, which is stored within the 3.0 file unless disabled from File->Preferences in Photoshop. If you disable this feature ("2.5 compatibility"), ThumbsPlus cannot read the file.

Why don't you support…?

With the profusion of graphic file formats in existence, we choose to include support for them based on the following criteria:
1. What customers request most often. We *do* listen to requests to support any file type. Please e-mail us or call us with your requests.
2. What is technically feasible. Many file types are proprietary and undocumented or poorly documented. Some are also so complex that we'd not be able to keep up with the application vendor — by the time we could support version 3, they'll have version 4!
3. What we are paid to support (for example, proprietary file types). We will be happy to provide a quote after format documentation and samples are provided.
4. What is royalty-free (with a couple of exceptions).

Why is OLE so slow?

Ask Microsoft, or the manufacturer of the specific application you're dealing with. In-process servers are generally much faster than executable-driven OLE servers, but are rare. For example, CorelDRAW starts up (and loads the entire program into memory) to serve an object. And, a new instance is started if Corel is already running, rather than simply calling the existing program to do the work.

Why can't I use my graphic import filters?

ThumbsPlus 3.0 only supports 32-bit import filters — if you have older sixteen-bit filters, ThumbsPlus will not be able to call them. Also, ThumbsPlus only supports filters that comply with the Aldus standard.

How many thumbnails can ThumbsPlus store in a single database?

There is no practical limit — the file structure is limited to about 4 billion thumbnails. We have tested with databases of over 100,000 thumbnails without experiencing significant speed problems. However, program startup is slower with large databases, and we'll improve on this in future releases.

Why does my database take more disk space in version 3 than it did in version 2?

While the actual thumbnail images are compressed exactly as much as before, ThumbsPlus stores more information for each thumbnail in the new version. Also, the indexes are larger because they are more robust (and faster) than the index in version two. Of course, if you make a database with larger thumbnails or more colors, the database will be larger for that also.
The new database format also automatically re-uses any disk space recovered from deleted or moved thumbnails; compacting (purging) should be needed much less frequently.

How can I call ThumbsPlus from my program (or database, spreadsheet, etc.)

ThumbsPlus currently only supports a rudimentary DDE command language (Open and Print commands); more extensive external control capabilities are planned using OLE 2 automation and possibly a callable API (DLL). We do not currently have an estimated date for implementation of either of these options.