

1 Everything

- 1.1 “Everything”是什么？
- 1.2 “Everything”建立数据库需要多长时间？
- 1.3 “Everything”能否搜索文件内容？
- 1.4 “Everything”是不是非常占用系统资源？
- 1.5 “Everything”能否监视文件系统更改？
- 1.6 “Everything”免费吗？
- 1.7 “Everything”不在运行的时候，它还能否监视文件系统更改？
- 1.8 “Everything”的系统要求？
- 1.9 如何把一个分区转化成 NTFS？

2 搜索

- 2.1 如何搜索？
- 2.2 如何运用布尔算子？
- 2.3 如何运用通配符？
- 2.4 如何运用正则表达式？
- 2.5 如何在搜索条件中包含空格？
- 2.6 如何搜索特定的文件类型？
- 2.7 如何指定搜索位置？

3 结果

- 3.1 如何跳转到搜索结果？

4 HTTP 服务器

- 4.1 如何同时搜索多个 HTTP 服务器？
- 4.2 如何通过一个 HTTP 服务器共享数据库？

5 定制

- 5.1 如何改变“Everything”按钮？
- 5.2 如何设置“Everything”而让它调用外部文件管理器？

6 故障排除

- 6.1 在 Windows Vista SP1 系统下运行，“Everything”需要管理员权限
- 6.2 没有搜索结果

1 Everything

1.1 “Everything” 是什么？

“Everything” 是一个运行于 Windows 系统，基于文件、文件夹名称的快速搜索引擎。“Everything” 在搜索之前就会把所用的文件和文件夹都列出来，这一点与 Windows 自带的搜索系统不一样，所以我们称之为“Everything”。

1.2 “Everything” 建立数据库需要多长时间？

“Everything” 搜索只基于文件和文件夹的名称，所以它创建数据库很快。一个刚安装完的 Windows XP SP2 系统(约 20,000 份文件)，需要一秒钟。索引一百万份文件则需要一分钟。

1.3 “Everything” 能否搜索文件内容？

不，不能搜索文件内容，“Everything” 搜索只基于文件和文件夹的名称。

1.4 “Everything” 是不是非常占用系统资源？

不，“Everything” 使用非常少的系统资源。一个刚安装完的 Windows XP SP2 系统(约 20,000 份文件)需要占用 3-5 mb 内存和不到 1 mb 的硬盘空间。一百万份文件大概需要 45 mb 内存和 5 mb 硬盘空间。

1.5 “Everything” 能否监视文件系统更改？

是的，“Everything” 能够监视文件系统改变。文件和文件夹名称的改变会实时地反映到“Everything” 数据库。

1.6 “Everything” 免费吗？

是的，“Everything” 是自由软件。

1.7 “Everything” 不在运行的时候，它还能否监视文件系统更改？

即使在“Everything” 没有运行的情况下，更改文件系统也不会有什么问题，因为“Everything” 在每次启动的时候会更新数据库。

1.8 “Everything” 的系统要求？

“Everything” 能够运行于 Windows 2000, XP, 2003 以及 Vista 下。记住：“Everything” 只能索引本地 NTFS 分区。

1.9 如何把一个分区转化成 NTFS？

参见 <http://support.microsoft.com/kb/307881>

2 搜索

2.1 如何搜索？

在搜索框中键入文件或文件夹名称的一部分，结果立刻呈现。

2.2 如何运用布尔算子？

AND 是缺省使用的布尔算子。

例如：如果要搜索 `foo` 和 `bar` 同时出现的文件：`foo bar`（注：在两个单词之间有空格）。

OR 用 `|` 表示。

例如：如果要搜索 `.jpg` 或 `.bmp` 文件：`.jpg | .bmp`（注：在 `|` 两侧有空格）。

2.3 如何运用通配符？

通配符 `*` 可以匹配任意长度和类型的字符。

例如：如果要搜索以 `e` 开头并且以 `g` 结尾的文件或文件夹：`e*g`。

通配符 `?` 可以匹配单个任意字符。

例如：如果要搜索扩展名为两个字符的文件：`*.??`。

2.4 如何运用正则表达式？

在使用之前，确保已经打开正则表达式选项（菜单 `Search->Enable Regex`）

<code> </code>	竖线表示或。例如： <code>gray grey</code> 能够匹配 <code>gray</code> 或 <code>grey</code> （注：在 <code> </code> 两侧没有空格）。
<code>()</code>	小括号用于确定范围。例如： <code>gr(a e)</code> 表示在 <code>gr</code> 和 <code>y</code> 之间有一个字母 <code>a</code> 或 <code>e</code> 。 <code>gr(a e)y</code> 将匹配 <code>gray</code> 或 <code>grey</code> ，与 <code>gray grey</code> 意义相同。
<code>?</code>	问号表示其前一个字符可能出现，也有可能不出现。例如： <code>colou?r</code> 将匹配 <code>color</code> 和 <code>colour</code> 。
<code>*</code>	星号表示其前一个字符可能不出现，也可能出现一次或多次。例如： <code>ab*c</code> 将匹配 <code>ac</code> 、 <code>abc</code> 、 <code>abbc</code> 、 <code>abbbc</code> 等等。
<code>+</code>	加号表示其前一个字符出现一次或多次。例如： <code>ab+c</code> 将匹配 <code>abc</code> 、 <code>abbc</code> 、 <code>abbbc</code> 等等，但不能匹配 <code>ac</code> 。
<code>.</code>	点号可用匹配任何单个非新行字符。（事实上，把哪个字符称之为新行却是不一定的，可能是编码特别或位置特别，但是可以肯定的是这个行中一定包含其他字符。）在 POSIX 括号表达式规则中，点号只匹配一个点。例如： <code>a.c</code> 可以匹配 <code>abc</code> 等等字符，但是 <code>[a.c]</code> 只匹配 <code>a</code> 、 <code>.</code> 、或 <code>c</code> 。
<code>[]</code>	中括号表示能够匹配其括号内出现的一个字符。例如： <code>[abc]</code> 能够匹配 <code>a</code> 、 <code>b</code> 或 <code>c</code> 。 <code>[a-z]</code> 将能够匹配 <code>a</code> 到 <code>z</code> 之间的任意一个字符。 <code>[abcx-z]</code> 能够匹配 <code>a</code> 、 <code>b</code> 、 <code>c</code> 、 <code>x</code> 、 <code>y</code> 或 <code>z</code> ，其也可以表示成 <code>[a-cx-z]</code> 。
<code>[^]</code>	<code>[^]</code> 表示可以匹配任何一个没有出现在其括号内的字符，与 <code>[]</code> 刚好相反。例如： <code>[^abc]</code> 能够匹配任何一个不是 <code>a</code> 、 <code>b</code> 、 <code>c</code> 的字符。 <code>[^a-z]</code> 能够匹配任何一个非小写字母之外的字符。
<code>^</code>	匹配名称开始的位置。在以行为基础的工具中，匹配任意行的开始位置。
<code>\$</code>	匹配名称结束的位置，或者匹配字符串结尾新行的结束位置。在以行为基础的工具中，匹配任意行的结束位置。
<code>{m,n}</code>	<code>{m,n}</code> 表示其前面的字符至少重复 <code>m</code> 次，但是重复次数少于 <code>n</code> 。例如： <code>a{3,5}</code> 匹配 <code>aaa</code> 、 <code>aaaa</code> 以及 <code>aaaaa</code> ，但是不能匹配 <code>aa</code> 或 <code>aaaaaa</code> 。此表达规则在一些老的版本中不可用。

2.5 如何在搜索条件中包含空格？

为了在搜索条件中包含空格，你可以使用双引号。例如：“foo bar”只能匹配 foo bar，而不能匹配 foobar、fooibar 等等。

2.6 如何搜索特定的文件类型？

为了搜索特定的文件类型，可以在搜索框中添加文件扩展名。

例如：搜索 mp3 文件可用 *.mp3。

如果想搜索多个文件类型，可以用 | 分割检索词。

例如：*.bmp | *.jpg 可用找到扩展名为 bmp 或 jpg 的文件。

2.7 如何指定搜索位置？

在检索条件中使用 \，可以指定查找位置。

例如：downloads\ .avi 表示在 downloads 文件夹中寻找所有 avi 文件。

也可用通过菜单 Search->Match Path 激活路径选项，这样可以把路径直接包含在搜索条件中。

例如：激活菜单 Search->Match Path 后，在 downloads 文件夹中需找所有 avi 文件可以表示为 downloads .avi。

3 结果

3.1 如何跳转到搜索结果？

选择需要的结果，鼠标右击，选择菜单。

4 HTTP 服务器

4.1 如何同时搜索多个 HTTP 服务器？

为了同时搜索多个 HTTP 服务器，需要打开 Allow multiple instances 选项，并且同时运行多个“Everything”窗口。

4.2 如何通过一个 HTTP 服务器共享数据库？

为了通过一个 HTTP 服务器共享数据文件，需要打开 Allow multiple instances 选项，并且同时运行多个“Everything”窗口。

5 定制

5.1 如何改变“Everything”按钮？

需要“Everything”1.2.0 beta 或更高版本。

复制你的图标文件，放进“everything”文件夹，并且重命名为“Everything.ico”。

5.2 如何设置“Everything”而让它调用外部文件管理器？

需要“Everything”1.2.0 beta 或更高版本。

退出“Everything”。

打开“Everything”安装文件夹中的 Everything.ini 文件。

添加如下两行到文件末尾。

```
open_folder_command=$exec("ExternalFileManager.exe" "%1")
```

```
open_folder_path_command=$exec("ExternalFileManager.exe" "$parent(%1)")
```

请用完整的路径名和文件名替换上两行中的 ExternalFileManager.exe。

阅读外部文件管理器的帮助，看是否需要其它的参数。

重启“Everything”。

6 故障排除

6.1 在 Windows Vista SP1 系统下运行，“Everything”需要管理员

权限

“Everything”需要高级别的硬盘读写权限，所以在 Vista SP1 系统下需要以管理员身份运行。

6.2 没有搜索结果

你的分区有可能是 FAT32 或 FAT16。

“Everything”只能检索 NTFS，对 FAT32、FAT16 不支持。

重要说明：这是根据“Everything”的英文 FAQ 翻译而来，

有不知所云的地方请参看原文。