

VPATH与vpath的区别与应用Linux认证考试 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/644/2021\\_2022\\_VPATH\\_E4\\_B8\\_8Ev\\_p\\_c103\\_644671.htm](https://www.100test.com/kao_ti2020/644/2021_2022_VPATH_E4_B8_8Ev_p_c103_644671.htm) 在Linux操作系统管理中，经常需要用到make命令来编译一些源代码。如内核升级，驱动程序安装等过程中，就需要用到这个命令。关于这个命令的使用技巧已经是老生常谈了。不过笔者这次要谈的是一个被大家所忽视的细节，即VPATH与vpath的区别，以及其在make命令中的应用。

一、让系统自动取寻找文件到依赖关系。在利用make命令来编译一些大型项目时，往往会涉及到大量的源文件。可能这个项目有多个管理员参加，为此存放在系统的不同目录中。此时在编译的时候，就需要指定文件的依赖关系。虽然可以在配制文件中指定，但是这不是最好的方法。因为一旦源代码改变了，就需要去更改配制文件。所以我们需要寻求一种更加简便的、更加灵活的方法。在Linux操作系统中，这个需求很容易实现。因为可以给make命令指定路径，然其在相关目录中去自动寻找文件的依赖关系。如此的话，只要存放源代码的目录没有改变，则就不用更改配制文件。所以采取这宗方法的话，相对来说可以简化make命令的编译工作。

二、通过VPATH来定义查找的路径。在Linux操作系统中，有一个PATH环境变量。当用户输入一个命令时，操作系统会现在这个环境变量指定的路径中查找命令。如果找不到的话，就会提示错误信息。在make命令中，也有一个类似参数，叫做VPATH。这个变量就是用来指定make命令查找的路径。如果用户没有指定这个VPATH 便量，则make命令只会在当前目录中去查找依赖文件和目标文件。但是如果用

户定义了这个变量，则这个命令首先会在当前的目录中查找依赖文件和目标文件。如果在当前目录下找不到的话，则还会自动从VPATH变量定义的路径中去查找。在这个变量中，可以定义多个路径。这个变量的定义规则跟PATH环境变量的定义规则有相同的地方，也有不同的地方。相同的地方是可以同时定义多个路径，而且路径之间都是利用英文状态下的冒号进行分割。两者的差异也很大。如在定义PATH变量的时候，要求采用的是绝对路径。而在定义VPATH变量时，即可以采用绝对路径，也可以采用相对路径。如../source，这个目录的定义方法(采用了相对路径)在定义PATH环境变量的时候，是不允许的。但是在这里定义VPATH变量的时候，确是允许的。所以在使用这个变量的时候，这个小细节需要注意。因为采用相对路径来表示的时候，相对来说要简便一点。而且将整个大的文件夹移动到另外一个位置上，原先定义的相对路径仍然有效。为此在能够使用相对路径来表示的，笔者还是建议大家使用相对路径。

### 三、通过vpath来定义查找的路径与匹配的模式。

利用VPATH变量来定义路径，大家可以看到其只是定义了一个查找的路径，而没有定义匹配的文件。如在某个目录中，除了源代码文件之外，还有不少的说明文档。如果在查找的时候，将所有的文件都判断一遍(包括一些说明文档)，那无疑会增加系统编译的时间。为此有时候，系统管理员不仅需要指定make命令查找的路径，而且还需要指定匹配的模式。即从符号哪些条件的文件中去查找。最常见的就是指定文件到扩展名。如此查找起来比较具有针对性，这个编译的速度也会提快许多。如现在系统管理员希望make命令查找指定路径下的以.h结尾的文件。要实现这个需求，

就需要用到另一个变量vpath。注意这里是小写，跟上面那个大写的VPATH虽然在功能上相似，不过功能要比其强一点。这个vpath变量与VPATH变量很相似，也都是用来指定make命令查找文件依赖关系的目录。不过这个变量更加的灵活。因为其除了可以指定文件查找的目录之外，还可以设置所匹配的模式。根据所带参数的不同，这个命令主要用来完成三项工作。一是什么参数都不带的情况下，就是一个vpath光棍司令，并不是表示采用当前的vpath变量。而是指清除所有已经设置好了的文件搜索目录。注意其清除vpath命令所设置的搜索路径，而不影响VPATH变量所设置的查找路径。系统管理员不要被它们的外表所蒙骗了。只要记住它们两个是不同的变量，虽然名字相同。二是带一个参数pattern，如vpath %.h。此时系统会清除符合模式的文件的搜索路径。如现在系统管理员可能会同时建立多个搜索策略，如.c、.h等等。而通过这种形式就只会删除符合模式的搜索路径。而不会影响到其它模式的搜索路径。三是带两个参数，如vpath %.h ..(这个命令表示在上级目录中查找.h文件)。两个参数(匹配模式与文件的搜索路径)都带齐的话，就表示在指定的目录中按规定的模式进行查找。不过需要注意的是，这里一个模式最好对应一条语句。也就是说，在一个vpath语句中定义一个匹配模式。如果要定义多个匹配模式的话，此时最好的办法是连续使用vpath语句来定义，以指定不同的搜索策略。不过需要注意的是，如果模式相同，第二次定义的时候就会更新原有的定义。也即是说，两者的关系是替换，而不是追加。为此同一个模式，如果需要在多个路径中查找的话，那么要么在一条语句中定义多个路径，要么就是将它们复制到同一个文件夹中

。一般最好将同一个模式的文件复制到同一个文件夹里，以方便管理。除非特别复杂的项目，才根据用途或者模块将它们分文件夹保存。不过在RedHat Linux系统中有点区别。在这个本地Linux操作系统中，可以相同模式的情况下定义多个路径。如`vpath %.h all`、`vpath %.h bll`、`vpath %.h cll`等等。此时后面对定义不会覆盖前面的定义。即三个语句是各自独立的，三个语句定义的变量都有效。当系统查找.h结尾的文件时，会根据`vpath`变量定义的时间顺序来查找。即先查找文件夹all，然后再查找bll，最后再查找cll。如果三个目录中有一个相同的文件，那么make命令就会以第一个目录中的文件为准。从以上的分析中可以看出，两个变量`vpath`与`VPATH`是两个不同的变量。其功能相似，都可以在寻找文件依赖关系时指定查找的路径。但是`vpath`其功能稍微强一点，可以按模式来查找，即更进一步的进行细分。为此，用户可以根据文件的多少、项目规模的大小来确定使用`VPATH`还是采用`vpath`。

四、使用这两个变量的注意事项。在使用这两个变量时，需要注意在变量中不用指定当前目录。在定义`PATH`环境变量中，大家可以看到其最后还加上了当前目录(.表示当前目录)。但是在定义这两个变量的时候，却不需要加上。因为其默认情况下，会先从当前目录下查找。只有当前目录下找不到对应的文件时，才会从这两个变量指定的路径中去查找。在存在多个路径的时候，其查找的时候是按照路径先后定义的顺序来的。linux认证网，加入收藏为此笔者要特别体系功能，`PATH`环境变量是把当前目录放在最后查找，而`VPATH`与`vpath`则是当前目录优先的。其次，对于`VPATH`与`vpath`两个变量来说，是可以采用相对路径来表示搜索路径的。而且

，无论从便利性还是从移植性来讲，采用相对路径会更加的合适一点。不过这也不是绝对的。需要根据项目实际的情况与用户的操作习惯来定。不过从笔者的个人倾向来说，还是喜欢采用相对路径。同时用户需要牢记的一点是，在定义PATH环境变量的时候，是不能够采用相对路径的，必须采用绝对路径。最后就是要注意这个两个变量的大小写。如果要在查找的过程中匹配模式话，那么就要采用小写的vpath，否则的话，则要采用大写的VPATH。如果张冠李戴的话，则可能会产生一些不利的影响。Make命令是系统管理员必须要掌握的一个源代码编译命令。因为不仅仅在编写应用程序的时候需要用到，有时候在对内核进行升级的时候，也需要用到这个命令。所以无论是程序员还是管理员，对于这个命令都要有很深的了解。100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)