# [yum方式安装的Apache目录详解和配置说明](http://www.cnblogs.com/carbon3/p/5635543.html)

在对httpd.conf文件进行解读之前,首先了解一下Redhat9中Apache服务器默认配置的一些基本信息:  
配置文件：/etc/httpd/conf/http.conf  
1)"/etc/httpd/conf  
主要存放了配置文件httpd.conf，这个是最重要的配置文件，Apache的所有主要权限和功能都在这个文件中进行了详细的设置。  
(2) "/etc/httpd/conf.d"  
里面存放的是一些额外的参数档，比如php.conf,或者一些自己设定的额外参数等信息。这个目录最大的好处就是可以自己设定自己的参数信息，比如我可以自己建立一个dl.conf，里面配置好相关参数，那么当apache启动的时候，这个文件会自动被读入到主要配置文件当中。这样的话当系统升级的时候，咱就不需要更动原本的配置文件，只要将自己设置的参数文件复制到正确的地点即可。  
(3) "/etc/httpd/logs"  
这里自然就是日志文件。  
(4) "/etc/httpd/modules"  
Apache模块的存放路径  
2./var/www：  
这里主要存放了  
(1)/var/www/html  
web服务器预设的首页文件存放目录。默认的主页是保存在其中的。此项默认页面可在httpd.conf中进行更改。这个后面会提及。   
其实安装好apache后，只要再次默认目录下设置好页面内容后就可以在浏览器中看到了。  
就好比我们刚刚安装好Apahche后，并没有过多设置什么，只需要在/var/www/html下如下设置即可：  
[root@linux102 html]# vi index.html //创建并编辑index.html  
hello world //页面内的内容  
[root@linux102 html]# service httpd restart  
Stopping httpd: [ OK ]  
Starting httpd: [ OK ]  
Ok，我们现在在浏览器中观察下是否显示我刚设置的内容：  


看 已经显示到了吧，其实到这里就是一个最最基本的web服务器，只要开启Apache，并且默认主页目录下有默认主页内容即可,但很多权限之类的设置却是在httpd.conf中设置。  
(2)/var/www/error/  
若主机因为设置错误，或者浏览器客户端要求的数据错误，在浏览器上显示的错误信息就是这个目录下的信息。  
(3)/var/www/icons/  
此目录是提供Apache默认的一些图标。  
(4)/var/www/cgi-bin/  
默认的一些可执行的CGI（网页程序）程序放置的目录  
3./var/log/httpd/  
默认的Apache日志文件保存在此  
4./usr/sbin/apachectl  
这个就不是目录了，此处的apachectl是文件，是apache的主要执行文件，这个执行文件其实是shell script，它可以主动检测系统上的一些设置值，好让您启动apache时更简单。简单的说就是apache 的一些管理工具  
5./usr/bin/htpasswd  
当我们想要登陆某些网页的时候，会提示输入账号和密码。而apache本身就提供一个最基本的密码保护方式，该密码的产生就是通过这个指令实现的。  
6./usr/sbin/httpd

服务器的根目录：/etc/httpd  
根文档目录：/var/www/html  
访问日志文件：/var/log/httpd/access\_log  
错误日志文件：/var/log/httpd/error\_log  
运行Apache的用户：apache  
运行Apache的组：apache  
端口：80  
模块存放路径：/usr/lib/httpd/moudules

prefork MPM运行方式的参数：  
StartServer 8  
MinSpareServices 5  
MaxSpareServer 20  
MaxClients 150  
MaxrequestsPerChild 1  
--------------------------------------------------------------------------------------------------------------------------  
下面是Redhat9 Linux下Apache服务httpd.conf的详细解读,自己对照了网上的一些中文翻译和Redhat9的Http.conf英文原文,针对网上的中文翻译有些修改和完善.我认为只是简单的看一下网上的翻译或是英文原文说明并没什么效果,何况网上的还有些小Miss,应该结合自己的理解和对照翻译,看自己理解的对不对,以加深对Apache配置信息的理解.这些整理时间上有点太急,肯定也会出现错误,请批评指正.:-)  
#  
# 基于 NCSA 服务的配置文件。  
#  
#这是Apache服务器主要配置文件。  
#它包含影响服务器运行的配置指令。  
#关于详细信息请参见<URL:http://httpd.ache.org/doc-2.0/>  
#  
#不要只是简单的阅读这些说明信息而不去理解它们做什么。  
#这里只是做了简单的说明，如果你没有参考在线文件，你就会被警告。  
#  
#这些配置指令被分为下面三个部分：  
#1. 控制整个Apache服务器行为的部分（即全局环境变量）  
#2. 定义主要或者默认服务参数的指令，也为所有虚拟主机提供默认的设置参数  
#3. 虚拟主机的设置参数,在一些Apache服务进程中配置不同的IP地址和主机名。  
#  
#配置和日志文件名：如果你指定的文件名以“/”开始（win32下以“dirver:/”），  
#服务器将使用绝对路径，如果文件名不是以“/”开始的，那么它将把ServerRoot  
#的值附加在文件名的前面，例如，对“logs/foo.log"，如果ServerRoot的值  
#为“/usr/local/apache2”，则该文件应为“/usr/local/apache2/logs/foo.log”

###第一区：全局环境参数  
#  
#这里设置的参数将影响整个Apache服务器的行为,  
#例如Apache能够处理的并发请求的数量或配置文件所在的位置等.  
  
#  
# Don't give away too much information about all the subcomponents  
# we are running. Comment out this line if you don't mind remote sites  
# finding out what major optional modules you are running // 问:这一段的中文意思是什么?请指教  
ServerTokens OS // 1) 当服务器响应主机头[header]信息时显示Apache的版本号和主机的操作系统;  
// 2) 如将"OS"改为"Prod",在出现错误页的时候不显示服务器操作系统的名称.

#  
#ServerRoot:目录树的根结点,指出服务器保存其  
#配置、出错信息和日志文件等的根目录。  
#  
#注意！如果你想要将它指定为NFS(或其它网络上的位置)  
#请一定要去阅读与LockFile有关的文档（可能在  
#<URL:http://httpd.apache.org/docs-2.0/mod/mpm\_common.html#lockfile>);  
#这样的话,您将节省大量的麻烦.  
#  
#在路径的结尾请不要添加斜线。  
#  
ServerRoot "/etc/httpd" // 设置服务器的根目录.

#ScoreBoardFile:用来保存内部服务进程信息的文件。  
#如果未指明（默认），scoreboard将被保存在一个匿名的共享内存段中，  
#并且它不能被第三方软件所使用。  
#如果指定了，要确保两个Apache不能使用同一个scoreboard文件，  
#这个scoreboard文件必须保存在本地磁盘上。  
#  
#ScoreBoardFile run/httpd.scoreboard

#  
#PidFile:当服务器启动时用于记录进程识别ID的文件。  
#  
PidFile run/httpd.pid // 设置运行Apache时使用的pidFile路径

#  
#Timeout:设置接收和发送前的超时秒数  
#  
Timeout 300 // 如果300秒没有收到或者送出任何数据就断开该连接

#  
#KeepAlive:是否允许保持连接功能（每个连接有多个请求），  
#设为"Off"则停用。  
#  
KeepAlive On // 使用保持连接的功能，如果为Off 则为不使用保持连接的功能，那样客户一次请求就只能响应一个文  
// 件，建议设为KeepAlive On

#  
#MaxKeepAliveRequests:在一个保持连接期间允许的最大请求数，  
#设为0表示无限制接入。  
#我们推荐你将其设为一个较大的值，以获得最高的性能.  
#  
MaxKeepAliveRequests 100 // 使用保持连接功能时，设置客户一次请求连接能响应的文件数最大上限为100

#  
#KeepAliveTimeout:同一连接同一客户端两个请求之间等待的时间秒数.  
#  
KeepAliveTimeout 15 // 使用保持连接功能时，如果两个相邻连接时间超过15秒，就断开连接。

##  
##Server-Pool大小设定（针对MPM）  
##

# prefork MPM  
# StartServers:启动时服务器的进程数  
# MinSpareServers:保有的备用进程的最小数目  
# MaxSpareServers:保有的备用进程的最大数目  
# MaxClients:服务器允许启动的最大进程数  
# MaxRequestsPerChild:一个服务进程允许的最大请求数  
<IfModule prefork.c> // 设置使用预生派(Prefork MPM)运行方式的参数，此方式是Redhat默认的方式  
StartServers 8 // 设置服务器启动时运行的进程数为8  
MinSpareServers 5 // 如果低于5个空闲子进程，就会创建新的子进程为客户提供服务  
MaxSpareServers 20 // 如果存在高于20个空闲子进程,就创建逐一删除的子进程来提高系统性能  
MaxClients 150 // 限制同一时间连接数不能超过150  
MaxRequestPerChild 1000 // 限制每个子进程在结束请求之前能处理的连接请求为1000  
</IfModule>

# worker MPM  
# StartServers:启动时的服务进程数目  
# MaxClients:允许同时连接的最大用户数目  
# MinSpareThreads:保有的最小工作线程数目  
# MaxSpareThreads:允许保有的最大工作线程数目  
# ThreadsPerChild:每个服务进程中的工作线程常数  
# MaxRequestsPerChild:服务进程中允许的最大请求数目  
<IfModule worker.c> // 设置使用工作者模式(worker MPM)运行方式的参数  
StartServers 2  
MaxClients 150   
MinSpareThreads 25  
MaxSpareThreads 75  
ThreadsPerChild 25  
MaxRequestsPerChild 0  
</IfModule>

# perchild MPM  
# NumServers:服务进程数量  
# StartThreads:每个服务进程中的起始线程数量  
# MinSpareThreads:保有的最小线程数量  
# MaxSpareThreads:保有的最大线程数量  
# MaxThreadsPerChild:每个服务进程允许的最大线程数  
# MaxRequestsPerChild:每个服务进程允许连接的最大数量  
<IfModule perchild.c> // 设置使用独立子进程(Perchild MPM)运行方式的参数  
NumServers 5  
StartThreads 5  
MinSpareThreads 5  
MaxSpareThreads 10  
MaxThreadsPerChild 20  
MaxRequestsPerChild 0  
</IfModule>

#  
# Listen:允许你将Apache服务绑定到指定的IP地址和端口上，以此取代默认值  
# 参见<VirtualHost>指令  
#  
# 改变如下命令使Apache只在指定的IP地址上监听，  
# 以防止它在IP地址(0.0.0.0)上监听  
#  
# Listen 12.34.56.78:80  
Listen 80 // 设置服务器的监听端口  
  
#  
# 配置文件从配置目录“/etc/httpd/conf.d ”加载  
#  
Include conf.dpublic\_html>  
# AllowOverride FileInfo AuthConfig Limit Indexes  
# Options MultiViews Indexes SymLinksIfOwnerMatch IncludeNoExec  
# <Limit GET POST OPTIONS PROPFIND>  
# Order allow,deny  
# Allow from all  
# </Limit>  
# <LimitExcept GET POST OPTIONS PROPFIND>  
# Order deny,allow  
# Deny from all  
# </LimitExcept>  
# </Directory>

#  
# DirectoryIndex:定义请求是一个目录时，Apache向用户提供服务的文件名  
#  
# index.html.var文件（一个类型映象文件）用于提供一个文档处理列表，  
# 出于同样的目的，也可以使用MultiViews选项，但是它会非常慢。  
#  
DirectoryIndex index.html index.html.var // 当访问服务器时依次查找index.htm index.html.var

#  
# AccessFileName:在每个目录中查询为目录提供附加配置指令的文件的文件名。  
# 参见AllowOverride指令。  
#  
AccessFileName .htaccess // 指定保护目录配置文件的名称

#  
# 下面的行防止.htaccess和.htpasswd文件被Web客户查看。  
#  
<Files ~ "^/.ht"> // 拒绝访问以.ht开头的文件，即保证.htaccess文件不被访问  
Order allow,deny  
Deny from all  
</Files>

#  
# Typeconfig:定义在哪里查询mime.types文件。  
#  
TypeConfig conf/mime.types // 指定负责处理MIME对应格式的配置文件的存放位置

#  
# DefaultType:定义当不能确定MIME类型时服务器提供的默认MIME类型。  
# 如果你的服务主要包含text或HTML文档，“text/plain”是一个好的选择；  
# 如果大多是二进制文档，诸如软件或图像，你应使用  
# “application/octer-stream”来防止浏览器象显示文本那样显示二进制文件。  
#  
DefaultType text/plain // 指定默认的MIME文件类型为纯文本或HTML文件

#  
# mod\_mime\_magic允许服务器从自己定义自己类型的文件中使用不同的线索（hints），  
# 这个MIMEMagicFile指令定义hints定义所在的文件。  
#  
<IfModule mod\_mime\_magic.c> // 当mod\_mime\_magic.c模块加载时，指定Magic信息码配置文件的存放位置  
# MIMEMagicFile /usr/share/magic.mime  
MIMEMagicFile conf/magic  
</IfModule>

#  
# HostnameLookups：指定记录用户端的名字还是IP地址，例如，本指令为on时  
# 记录主机名，如www.apache.org；为off时记录IP地址，204.62.129.132。  
# 默认值为off，这要比设为on好得多，因为如果设为on则每个用户端请求都将会  
# 至少造成对 nameserver 进行一次查询。  
#  
HostnameLookups Off // 只记录连接Apache服务器的IP地址，而不记录主机名  
#  
# ErrorLog:错误日志文件定位。  
# 如果你没有在<VirtualHost>内定义ErrorLog指令，这个虚拟主机的错误信息  
# 将记录在这里。如果你在那儿定义了ErrorLog，这些错误信息将记录在你所  
# 定义的文件里，而不是这儿定义的文件。  
#  
ErrorLog logs/error\_log // 指定错误日志的存放位置

#  
# LogLevel:控制记录在错误日志文件中的日志信息数量。  
# 可能的值包括：debug，info，notice，warn，error，crit，alert，emerg。  
#  
LogLevel warn // 指定记录的错误信息的详细等级为warn 级别

#  
# 下面的指令为CustomLog指令定义格式别名。  
#  
LogFormat "%h %l %u %t /"%r/" %>s %b /"%{Referer}i/" /"%{User-Agent}i/"" combined // 定义4种日志记录格式  
LogFormat "%h %l %u %t /"%r/" %>s %b" common  
LogFormat "%{Referer}i -> %U" referer  
LogFormat "%{User-agent}i" agent

#  
# 指定接入日志文件的定位和格式（一般日志格式）。  
# 如果你没有在<VirtualHost>内定义这个指令，传输信息将记录在这里，  
# 如果你定义了这个指令，则记录在你指定的位置，而不是这儿定义的位置。  
#  
# CustomLog logs/access\_log common  
CustomLog logs/access\_log combined // 指定访问日志的记录格式为combined(混合型)，并指定访问日志的存放位置

#  
# 如果你想要记录agent和referer信息，可以使用下面的指令  
#  
# CustomLog logs/referer\_log referer  
# CustomLog logs/agent\_log agent

#  
# 如果你想要使用一个文件记录access，agent和referer信息，  
# 你可以如下定义这个指令：  
#  
# CustomLog logs/access\_log combined

#  
# 随意的添加包含服务器版本和虚拟主机名字一行信息到server-generated输出页中  
# （内部错误文档，FTP目录列表，mod\_status和mod\_info输出等等，除了CGI错误  
# 或自定义的错误文档以外）。  
# 设为“EMail”将包含一个指向ServerAdmin的mailto:连接。  
# 可以为如下值：On | Off | EMail  
#  
ServerSignature On // 设置Apache自己产生的页面中使用Apache服务器版本的签名

#  
# Aliases:在这时添加你需要的别名，格式如下：  
# Alias 别名真实名  
#  
# 注意，如果你在别名的未尾包含了“/”，那么在URL中也需要包含“/”。  
# 因此，“/icons”不是这个示例中的别名。  
# 如果别名中以“/”结尾，那么真实名也必须以“/”结尾，  
# 如果别名中省略了结尾的“/”，那么真实名也必须省略。  
#  
# 我们使用别名“/icons/”来表示FancyIndexed目录列表，如果你不使用、  
# FancyIndexing，你可以注释掉它。  
#  
Alias /icons/ "/var/www/icons/" // 设置内容协商目录的访问别名

<Directory "/var/www/icons"> // 设置/var/www/icons目录的访问权限   
Options Indexes MultiViews // MultiViews： 使用内容协商功能决定被发送的网页性质  
AllowOverride None  
Order allow,deny  
Allow from all  
</Directory>

#  
# 这将改变ServerRoot/manual。这个别名提供了手册页所在的位置，  
# 即使你改变了你的DocumentRoot。如果你对有无手册页并不在意的话，  
# 你可以注释掉它。  
#  
Alias /manual "/var/www/manual" // 设置Apache手册的访问别名

<Directory "/var/www/manual"> // 设置/var/www/manual 目录的访问权限  
Options Indexes FollowSymLinks MultiViews  
AllowOverride None  
Order allow,deny  
Allow from all  
</Directory>  
  
<IfModule mod\_dav\_fs.c> // 指定DAV加锁数据库文件的存放位置  
# Location of the WebDAV lock database.  
DAVLockDB /var/lib/dav/lockdb  
</IfModule>

#  
# ScriptAlias:指定包含服务脚本的目录。  
# ScriptAliases 本质上与Aliases一样，除了这里的文档在请求时做为程序处理处理以外。  
# 尾部的“/”规则与Alias一样  
#  
ScriptAlias /cgi-bin/ "/var/www/cgi-bin/" // 设置CGI目录的访问别名

<IfModule mod\_cgid.c> // 由于Redhat9中不使用worker MPM运行方式，所以不加载mod\_cigd.c模块  
#  
# 添加mod\_cgid.c设置，mod\_cgid提供使用cgid进行通讯的UNIX套接字的  
# 脚本接口路径。  
#  
Scriptsock run/httpd.cgid  
</IfModule>

#  
# 将"/var/www/cgi-bin"改为你的ScriptAliased指定的CGI目录，  
# 如果你配置了的话。  
#  
<Directory "/var/www/cgi-bin"> // 设置CGI目录的访问权限  
AllowOverride None  
Options None  
Order allow,deny  
Allow from all  
</Directory>

#  
# Redirect允许你告诉客户端使用存在于服务器名字空间中的文档，  
# 而不是现在的，这帮助客户定位那些改变了位置的文档。  
# 例如：  
# Redirect permanent /foo http://www.example.com/bar

#  
# 控制server-generated目录列表显示的指令  
#  
#  
# FancyIndexing is whether you want fancy directory indexing or standard.  
# VersionSort is whether files containing version numbers should be  
# compared in the natural way, so that `apache-1.3.9.tar' is placed before  
# `apache-1.3.12.tar'. // 问:这一段的英文意思是什么?:)  
#  
IndexOptions FancyIndexing VersionSort NameWidth=\* // FancyIndexing:对每种类型的文件前加上一个小图标以示区别  
// VersionSort:对同一软件的多个版本进行排序   
// NameWidth=\*:文件名字段自动适应当前目录下最长文件名

#  
# AddIcon\* 指令告诉服务器不同扩展名的图象文件如何显示，  
# 只适用于FancyIndexed指令  
#  
AddIconByEncoding (CMP,/icons/compressed.gif) x-compress x-gzip // 当使用FancyIndexing VersionSort 之后，配置下面的参数  
// 用于告知服务器在碰到不同的文件类型或扩展名时采用  
AddIconByType (TXT,/icons/text.gif) text/\* // 指定的格式,辨别文件并显示对应的图标   
AddIconByType (IMG,/icons/image2.gif) image/\*  
AddIconByType (SND,/icons/sound2.gif) audio/\*  
AddIconByType (VID,/icons/movie.gif) video/\*

AddIcon /icons/binary.gif .bin .exe  
AddIcon /icons/binhex.gif .hqx  
AddIcon /icons/tar.gif .tar  
AddIcon /icons/world2.gif .wrl .wrl.gz .vrml .vrm .iv  
AddIcon /icons/compressed.gif .Z .z .tgz .gz .zip  
AddIcon /icons/a.gif .ps .ai .eps  
AddIcon /icons/layout.gif .html .shtml .htm .pdf  
AddIcon /icons/text.gif .txt  
AddIcon /icons/c.gif .c  
AddIcon /icons/p.gif .pl .py  
AddIcon /icons/f.gif .for  
AddIcon /icons/dvi.gif .dvi  
AddIcon /icons/uuencoded.gif .uu  
AddIcon /icons/script.gif .conf .sh .shar .csh .ksh .tcl  
AddIcon /icons/tex.gif .tex  
AddIcon /icons/bomb.gif core

AddIcon /icons/back.gif ..  
AddIcon /icons/hand.right.gif README  
AddIcon /icons/folder.gif ^^DIRECTORY^^  
AddIcon /icons/blank.gif ^^BLANKICON^^

#  
# DefaultIcon 为那些没有显式定义图标的文件提供处理  
#  
DefaultIcon /icons/unknown.gif // 当使用FancyIndexing VersionSort 之后，且无法识别文件类型时，显示此处定义的图标

#  
# AddDescription允许你在server-generated索引后放置一个简短的说明。  
# 只对FancyIndexed指令有效。  
# 格式：AddDescription "说明" 文件名  
#  
# AddDescription "GZIP compressed document" .gz  
# AddDescription "tar archive" .tar  
# AddDescription "GZIP compressed tar archive" .tgz

#  
# ReadmeName指定服务器默认查找的README文件的名字，并添加到目录列表中  
#  
# HeaderName指定目录列表前缀文件的文件名  
ReadmeName README.html // 当服务器自动列出目录列表时，在所生成的页面之后显示README.html的内容  
HeaderName HEADER.html // 当服务器自动列出目录列表时，在所生成的页面之前显示HEADER.html的内容

#  
# IndexIgnore指定目录索引忽略并且不包含在列表中的文件名集合，  
# 支持shell类型的通配符。  
#  
IndexIgnore .??\* \*~ \*# HEADER\* README\* RCS CVS \*,v \*,t

#  
# AddEncoding允许你在信息传送中使用(Mosaic/X 2.1+)解压缩信息，  
# 注意：不是所有的浏览器都支持这个选项。  
# 尽管名字相似，但是下列的指令与上面的FancyIndexing定制指令不同。  
#  
AddEncoding x-compress Z // 设置在线浏览器可以实时解压.Z .gz .tgz类型文件，但并非所有浏览器都支持  
AddEncoding x-gzip gz tgz

#  
# DefaultLanguage和AddLanguage允许你指定文档的语言。  
# 这使你可以让用户用容易理解的语言浏览文档。  
#  
# 指定默认的语言，这意味着所有没有指定语言的包都将使用该语言。  
# 多数情况下，你也许并不想设置它，除非你确信这样做是正确的。  
# 通常，不使用确定的语言比使用错误的语言要好。  
#  
# DefaultLanguage nl  
#  
# 注意1：作为语言关键字的词缀毫无疑问是不能一样的--采用波兰  
# 文的文档（网络标准语言代码是pl）将希望使用“AddLanguage pl .po”  
# 来避免与perl脚本的一般词缀产生二义性。  
#  
# 注意2: 下面的例子举例说明在一些范例中语言的二字符缩写与它的国家  
# 的二字符缩写不相同,例如 “Danmark/dk” 和 “Danmark/da” 的比较.  
#  
# 注意3: 在 “ltz” 的情况下我们使用三字符词缀，违犯了 RFC的规定，  
# 运行中将修复它并使用RFC1766标准取得参考数据。  
#  
# Danish (da) - Dutch (nl) - English (en) - Estonian (et)  
# French (fr) - German (de) - Greek-Modern (el)  
# Italian (it) - Norwegian (no) - Norwegian Nynorsk (nn) - Korean (kr)  
# Portugese (pt) - Luxembourgeois\* (ltz)  
# Spanish (es) - Swedish (sv) - Catalan (ca) - Czech(cz)  
# Polish (pl) - Brazilian Portuguese (pt-br) - Japanese (ja)  
# Russian (ru) - Croatian (hr)  
#  
AddLanguage da .dk // 设置网页内容语言种类，(浏览器要启用内容协商)，对于中文网页，此项无实际意义  
AddLanguage nl .nl  
AddLanguage en .en  
AddLanguage et .et  
AddLanguage fr .fr  
AddLanguage de .de  
AddLanguage he .he  
AddLanguage el .el  
AddLanguage it .it  
AddLanguage ja .ja  
AddLanguage pl .po  
AddLanguage kr .kr  
AddLanguage pt .pt  
AddLanguage nn .nn  
AddLanguage no .no  
AddLanguage pt-br .pt-br  
AddLanguage ltz .ltz  
AddLanguage ca .ca  
AddLanguage es .es  
AddLanguage sv .se  
AddLanguage cz .cz  
AddLanguage ru .ru  
AddLanguage tw .tw  
AddLanguage zh-tw .tw  
AddLanguage hr .hr

# LanguagePriority允许你在会话过程中优先使用一些语言。  
#  
# 以优先次序递减的方式列出它们。我们或多或少地采用按字母排列顺序的方式  
# 排列它们。也许你想要改变这个顺序。  
LanguagePriority en da nl et fr de el it ja kr no pl pt pt-br ltz ca es sv tw // 当启用内容协商时，设置语言的先后顺序

#  
# ForceLanguagePriority 允许你为MULTIPLE CHOICES(Prefer)[在通讯的情况下]  
# 或NOT ACCEPTABLE(Fallback)[没有可接受的语言匹配的情况]提供一个结果页。  
#  
ForceLanguagePriority Prefer Fallback // Prefer：当有多种语言可以匹配时，使用LanguagePriority 列表的第一项  
// Fallback：当没有语言可以匹配时，使用LanguagePriority 列表的第一项

#  
# 为发送出的所有页指定默认的字符集，这总是一个好主意，并且为你的  
# web站点的国际化打开了大门，这不正是你曾经想要的吗。同样地，指定  
# 默认字符集有一些小的损害，如一个使用iso-8859-1(latin1)标准命令  
# 的页面，除非以别的方式指定例如你仅仅以显式方式声明它。  
# 也有一些与那些总是鼓励你使用默认字符集的javascropt和URL语法有关  
# 的浏览器安全原因。  
#  
AddDefaultCharset ISO-8859-1 // 设置默认字符集(要设置成为简体中文,把"ISO-8859-1"改为"GB2312")

#  
# 一般以文件扩展名的方式使用字符集。也许你想要避免与语言扩展发生  
# 碰撞，除非你在每次改变后都做了很好的测试。  
# 参见http://www.iana.org/assignments/character-sets以取得字符集  
# 的名字列表和它们各自的RFCs。  
#  
AddCharset ISO-8859-1 .iso8859-1 .latin1 // 设置各种字符集  
AddCharset ISO-8859-2 .iso8859-2 .latin2 .cen  
AddCharset ISO-8859-3 .iso8859-3 .latin3  
AddCharset ISO-8859-4 .iso8859-4 .latin4  
AddCharset ISO-8859-5 .iso8859-5 .latin5 .cyr .iso-ru  
AddCharset ISO-8859-6 .iso8859-6 .latin6 .arb  
AddCharset ISO-8859-7 .iso8859-7 .latin7 .grk  
AddCharset ISO-8859-8 .iso8859-8 .latin8 .heb  
AddCharset ISO-8859-9 .iso8859-9 .latin9 .trk  
AddCharset ISO-2022-JP .iso2022-jp .jis  
AddCharset ISO-2022-KR .iso2022-kr .kis  
AddCharset ISO-2022-CN .iso2022-cn .cis  
AddCharset Big5 .Big5 .big5  
# 对于俄语，使用了多个字符集（如何使用主要依靠客户端）：  
AddCharset WINDOWS-1251 .cp-1251 .win-1251  
AddCharset CP866 .cp866  
AddCharset KOI8-r .koi8-r .koi8-ru  
AddCharset KOI8-ru .koi8-uk .ua  
AddCharset ISO-10646-UCS-2 .ucs2  
AddCharset ISO-10646-UCS-4 .ucs4  
AddCharset UTF-8 .utf8

# 下面的字符集没有映射到一个特定的标准(iso)上，但是它们在浏览器  
# 中被广泛的支持。注意那些大写字母。  
# (它不应该,但是它是为兼容一些浏览器而做)  
#  
# 参见http://www.iana.org/assianments/character-sets以取得  
# 它们的列表。但是浏览器支持较少。  
#  
AddCharset GB2312 .gb2312 .gb  
AddCharset utf-7 .utf7  
AddCharset utf-8 .utf8  
AddCharset big5 .big5 .b5  
AddCharset EUC-TW .euc-tw  
AddCharset EUC-JP .euc-jp  
AddCharset EUC-KR .euc-kr  
AddCharset shift\_jis .sjis

#  
# AddType允许你为指定的文件类型添加或覆盖mime.types文件中配置的MIME  
#  
AddType application/x-tar .tgz // 增加新的MIME类型(避免用户编辑/etc/mime.types)

#  
# AddHandler允许你映射确定的文件扩展名到“handlers”:  
# 与文件类型无关的行为。这既能编译到服务器中也可以添加到Action指令  
# 中（看下面）。  
#  
# 为了在ScriptAliased指令指定的以外使用CGI脚本：  
#（要使它可用，你还需要在Options中添加“ExecCGI”。  
#  
#AddHandler cgi-script .cgi

#  
# 对于那些包含他们自己的HTTP头的文件  
#  
# AddHandler send-as-is asis

#  
# 对于server-parsed imagemap文件：  
#  
AddHandler imap-file map // 设置Apache对某些扩展名的处理方式

#  
# 对于类型映像:(转移资源)  
#(这是默认的设定以允许Apache的“It Worked”页能多种语言分发)。  
#  
AddHandler type-map var

#  
# 过滤器允许你在将它发送到客户端前进行处理。  
#  
# 为了在服务器端分析包含(SSI)的.shtml文档：  
# （要执行这个指令，你还需要在Options指令中添加“Includes”。）  
#  
AddOutputFilter INCLUDES .shtml // 使用过虑器执行SSI

#  
# Action让你定义当调用匹配的媒体文件时将要执行的脚本。这将减少  
# 那些经常使用的CGI脚本的URL路径名的重复输入。  
# 格式：Action media/type /cgi-script/location  
# 格式：Action handler-name /cgi-script/location  
#

#  
# 可配置的错误应答有三种风格：  
# 1）plain text 2)local redirects 3) external redirects  
#  
# 一些示例：  
# ErrorDocument 500 "The server made a boo boo."  
# ErrorDocument 404 /missing.html  
# ErrorDocument 404 "/cgi-bin/missing\_handler.pl"  
# ErrorDocument 402 http://www.example.com/subscription\_info.html  
#

#  
# 综合应用这些指令，我们可以创建一个国际化的出错应答。  
#  
# 我们使用Alias来重定向任意/error/HTTP\_<error>.html.var应答到  
# 我们的多语言错误消息集合。使用正确的文本替代它。  
#  
# 通过加入下面的行，你就能够改变这些消息的显示，而不必改变  
# HTTP\_<error>.html.var文件。  
#  
# Alias /error/include/ "/your/include/path/"  
#  
# 以将/var/www/error/include/下的文件拷贝到/your/inclue/path/下  
# 开始，你可以创建你自己的文件集合，甚至是基于每个虚拟主机的。  
#   
Alias /error/ "/var/www/error/" // 设置错误页面目录的别名

<IfModule mod\_negotiation.c>   
<IfModule mod\_include.c>  
<Directory "/var/www/error"> // 设置/var/www/error目录的访问权限  
AllowOverride None  
Options IncludesNoExec  
AddOutputFilter Includes html  
AddHandler type-map var  
Order allow,deny  
Allow from all  
LanguagePriority en es de fr  
ForceLanguagePriority Prefer Fallback  
</Directory>  
// 设置错误页面输出  
ErrorDocument 400 /error/HTTP\_BAD\_REQUEST.html.var  
ErrorDocument 401 /error/HTTP\_UNAUTHORIZED.html.var  
ErrorDocument 403 /error/HTTP\_FORBIDDEN.html.var  
ErrorDocument 404 /error/HTTP\_NOT\_FOUND.html.var  
ErrorDocument 405 /error/HTTP\_METHOD\_NOT\_ALLOWED.html.var  
ErrorDocument 408 /error/HTTP\_REQUEST\_TIME\_OUT.html.var  
ErrorDocument 410 /error/HTTP\_GONE.html.var  
ErrorDocument 411 /error/HTTP\_LENGTH\_REQUIRED.html.var  
ErrorDocument 412 /error/HTTP\_PRECONDITION\_FAILED.html.var  
ErrorDocument 413 /error/HTTP\_REQUEST\_ENTITY\_TOO\_LARGE.html.var  
ErrorDocument 414 /error/HTTP\_REQUEST\_URI\_TOO\_LARGE.html.var  
ErrorDocument 415 /error/HTTP\_SERVICE\_UNAVAILABLE.html.var  
ErrorDocument 500 /error/HTTP\_INTERNAL\_SERVER\_ERROR.html.var  
ErrorDocument 501 /error/HTTP\_NOT\_IMPLEMENTED.html.var  
ErrorDocument 502 /error/HTTP\_BAD\_GATEWAY.html.var  
ErrorDocument 503 /error/HTTP\_SERVICE\_UNAVAILABLE.html.var  
ErrorDocument 506 /error/HTTP\_VARIANT\_ALSO\_VARIES.html.var

</IfModule>  
</IfModule>

#  
# 下面的命令更改标准的HTTP应答行为以处理己知的浏览器问题。  
#  
// 设置浏览器匹配  
BrowserMatch "Mozilla/2" nokeepalive   
BrowserMatch "MSIE 4/.0b2;" nokeepalive downgrade-1.0 force-response-1.0  
BrowserMatch "RealPlayer 4/.0" force-response-1.0  
BrowserMatch "Java/1/.0" force-response-1.0  
BrowserMatch "JDK/1/.0" force-response-1.0

#  
# 下面命令关闭对那些没有尾部“/”的目录的非GET请求的重定向，  
# 这些命令修复了微软的采用DAV方法不能正确处理重定向的WEB文件夹的问题。  
# Apple下的DAV文件系统和Gnome下的VFS对DAV的支持也是采用这样的方法  
# 进行处理的。  
#  
BrowserMatch "Microsoft Data Access Internet Publishing Provider" redirect-carefully  
BrowserMatch "^WebDrive" redirect-carefully

#  
# 允许你使用URL：http://servername/server-status来通过mod\_status生  
# 成并报告服务器状态信息。改变.example.com为你自己的域名。  
#  
#<Location /server-status>  
# SetHandler server-status  
# Order deny,allow  
# Deny from all  
# Allow from .your-domain.com  
#</Location>

#  
# 允许使用URL:http://servername/server-info来远程报告服务器配置信息  
# （需要mod\_info.c支持）。改变“.example.com”为你自己的域名。  
#  
#<Location /server-info>  
# SetHandler server-info  
# Order deny,allow  
# Deny from all  
# Allow from .your-domain.com  
#</Location>

#  
# 代理服务器命令，去掉下面的行使代理服务可用。  
#  
#<IfModule mod\_proxy.c>  
#ProxyRequests On  
#  
#<Proxy \*>  
# Order deny,allow  
# Deny from all  
# Allow from .your-domain.com  
#</Proxy>

#  
# 安装或关闭HTTP/1.1“通道”头处理。  
# （“Full”添加服务器版本信息，“Block”移掉所有输出“通道”头信息。  
# 可以设为下面各选项之一：Off | On | Full | Block  
#  
# ProxyVia On

# 最好为代理服务安装高速缓冲，去掉下面几行的注释符号：  
# （没有CacheRoot则不缓冲）  
#  
#CacheRoot "/etc/httpd/proxy"  
#CacheSize 5  
#CacheGcInterval 4  
#CacheMaxExpire 24  
#CacheLastModifiedFactor 0.1  
#CacheDefaultExpire 1  
#NoCache a-domain.com another-domain.edu joes.garage-sale.com

# </IfModule>  
# 代理命令结束。

### 第三区：虚拟主机  
#  
# VirtualHost:你可以通过设置虚拟主机容器以实现在你的主机上保有多个  
# 域名/主机名。大多数配置信息只使用基于名字的虚拟主机，因此服务器  
# 不必担心IP地址的问题，下面的命令以\*号代替虚拟主机名。  
#  
# 在你试着配置你的虚拟主机以前，请参见  
# URL:http://httpd.apache.org/docs-2.0/vhosts/>以取得更多的信息。  
#  
# 你可以使用命令行选项“-S”来检验你的虚拟主机配置。

#  
# 使用基于名字的虚拟主机。  
#  
# NameVirtualHost \*

#  
# 虚拟主机示例：  
# 几乎所有的Apache命令都可以在虚拟主机容器中使用。  
# 第一个虚拟主机区是用于向服务名未知的请求进行应答的配置。  
#  
#<VirtualHost \*>  
# ServerAdmin webmaster@dummy-host.example.com  
# DocumentRoot /www/docs/dummy-host.example.com  
# ServerName dummy-host.example.com  
# ErrorLog logs/dummy-host.example.com-error\_log  
# CustomLog logs/dummy-host.example.com-access\_log common  
#</VirtualHost>