## zjufirefly



文章搜索

文章分类
C++ (14)
其他 (5)
链接 (13)
linux (13)
shell (7)
操作 (11)

文章存档

2014年12月 (8)
2014年11月 (7)
2014年10月 (8)
2014年09月 (6)
2014年08月 (4)

展开

 関读排行
 (327)

 git操作命令
 (287)

 blktrace
 (287)

 mysql常用命令
 (280)

 python
 (270)

 cgroups
 (262)

 linux下如何配置openvpn
 (260)

RSS 订阅 万新文章 目录视图 ₩ 摘要视图 管理博客 2015年4月微软MVP申请 《京东技术解密》有奖试读,礼品大放送 "我的2014"年度征文活动火爆开启 CSDN 2014博客 makefile 分类: 其他 链接 2014-12-22 19:33 107人阅读 评论(0) 收藏 编辑 删除 makefile wildcard 头文件依赖 http://blog.vjeux.com/category/makefile http://scottmcpeak.com/autodepend/autodepend.html http://www.freezhongzi.info/?p=216 依赖解释 http://maskray.me/blog/2011-08-11-generate-dependency-in-makefile 层级makefile http://blog.csdn.net/hmsiwtv/article/details/20905177 静态库 http://www.cnblogs.com/itech/archive/2009/09/08/1562292.html http://www.360doc.com/content/09/0421/19/36491\_3216558.shtml

makefile 处理 通配符展开 srcs:=\$(wildcard \*.c) 后缀替换 \$(source:.d=.c) 增加前缀

\$(source:%=pre%)

增加后缀

\$(source:%=%suf)

转义

当在Makefile中需要使用字符"#"时,可以使用\#来实现 当在Makefile中需要使用字符"\$"时,可以使用\$\$来实现 统配符\*使用\\*转义

在模式规则中字符'%'可以用前面加反斜杠"\"方法引用。

查找的makefile文件名顺序为

GNUmakefile , makefile , Makefile .

使用**include**标示符包含的文件文件查找顺序 当前文件夹

vim-ctags-taglist-netrw	(249)
性能测试命令字段解释	(213)
正则表达式	(211)
SystemTap	(195)

评论排行 为什么需要auto\_ptr\_ref (1) valgrind内存检查 (0) (0) cgroups linux下如何配置openvpn (0) git操作命令 (0) thrift (0) (0) Apache开源软件 (0) vim-ctags-taglist-netrw vim常用配置 (0) shell按行读取字符串,并 (0)

#### 推荐文章

- \* 挣扎与彷徨--我的2014
- \* 校招回忆录---小米篇
- \* Android UI-自定义日历控件
- \* 30岁程序员回顾人生、展望未来
- \* 2014年终总结, 我决定要实现的
- \* 2014年经总结,我快走安头观日 三个目标
- \* Android 启动问题——黑屏 死机解决方法

最新评论

为什么需要auto\_ptr\_ref zjufirefly: explicit auto\_ptr\_ref(\_Tp1\* \_\_p): \_M\_ptr(\_\_p) { }... 命令行选项-I或者--include-dir指定的目录

/usr/gnu/include

/usr/local/include

/usr/include

-include, sinclude来忽略由于包含文件不存在或者无法创建时的错误提示

共享库查找路径

当前文件夹

VPATH vpath"指定的搜索目录。

/lib /usr/lib /usr/local/lib

#### 多个makefile读取顺序

make在执行时,首先读取的是环境变量"MAKEFILES"所指定的文件列表,之后才是工作目录下的makefile文件,"include"所指定的文件是在make发现此关键字的时、暂停正在读取的文件而转去读取"include"所指定的文件。

make -t 更新时间戳

make -n 打印命令

所有匹配模式

#sample GNUmakefile

foo:

frobnicate > foo

%: force

@\$(MAKE) -f Makefile \$@

force:;

#### make如何解析makefile文件

第一阶段:读取所有的makefile文件,内建所有的变量、明确规则和隐含规则,并建立所有目标和依赖之间的依赖关系结构链表。

在第二阶段:根据第一阶段已经建立的依赖关系结构链表决定哪些目标需要更新,并使用对应的规则来重建这些目标。

在make执行的第一阶段中如果变量和函数被展开,那么称此展开是"立即"的,其他的展开称之为"延后"的。

变量及规则的展开

IMMEDIATE = DEFERRED

IMMEDIATE ?= DEFERRED

IMMEDIATE := IMMEDIATE

IMMEDIATE += DEFERRED or IMMEDIATE

define IMMEDIATE

**DEFERRED** 

Fndef

ifdeq ifeq是立即的

所有的规则在make执行时,都按照如下的模式展开:

IMMEDIATE: IMMEDIATE; DEFERRED

DEFERRED

规则中目标和依赖如果引用其他的变量,则被立即展开。而规则的命令行中的变量引用会被延后展开。

make的执行过程如下:

- 1. 依次读取变量MAKEFILES定义的makefile文件列表
- 2. 读取工作目录下的makefile文件(根据命名的查找顺序GNUmakefile,makefile,Makefile首先找到那个就读取那个)
  - 3. 依次读取工作目录makefile文件中使用指示符"include"包含的文件
  - 4. 查找重建所有已读取的makefile文件的规则(如果存在一个目标是当前读取的某一个makefile文件,则执行

此规则重建此makefile文件,完成以后从第一步开始重新执行)

- 5. 初始化变量值并展开那些需要立即展开的变量和函数并根据预设条件确定执行分支
- 6. 根据"终极目标"以及其他目标的依赖关系建立依赖关系链表
- **7**. 执行除"终极目标"以外的所有的目标的规则(规则中如果依赖文件中任一个文件的时间戳比目标文件新,则使用规则所定义的命令重建目标文件)
  - 8. 执行"终极目标"所在的规则

### gcc编译详解

http://www.cnblogs.com/azraelly/archive/2012/07/07/2580839.html

Gcc –E hello.c –o hello.i 预处理,包含头文件,宏展开

Gcc - S hello.i - o hello.s 汇编,生成可阅读的汇编代码

Gcc –c hello.s –o hello.o 生成二进制文件 Gcc hello.o –o hello 生成可执行文件

上一篇 iSCSI

下一篇 makefile范例

顶踩

主题推荐 makefile 二进制 工作 链表 汇编

#### 猜你在找

高通平台android环境搭建编译系统引导流程分析

android linux 基础知识总结

linux Android基础知识总结

C++工程实践经验

7 zImage的生成和加载

android linux 最全的基础知识总结 android linux 基础知识总结 透过源码领悟GCC到底在干些什么收集整理 shell学习资料shell十三问

腾讯后台开发面试题及答案

# ● 万网 买域名只要29元 注册免费领取20元代金券 (VR.com/.net/:cn

查看评论	
暂无评论	
发表评论	
用 户 名: 评论内容:	zjufirefly
7710774	

\*以上用户言论只代表其个人观点,不代表CSDN网站的观点或立场

提交

核心技术类目

全部主題 HaJop AWS 移动游戏 Java Android iOS Swift 智能硬件 Docker OpenStack
VPN Spark ERP IE10 Eclipse CRM JavaScript 数据库 Ubuntu NFC WAP jQuery
BI HTML5 Spring Apache .NET API HTML SDK IIS Fedora XML LBS Unity
Splashtop UML components Windows Mobile Rails QEMU KDE Cassandra CloudStack
FTC coremail OPhone CouchBase 云计算 iOS6 Rackspace Web App SpringSide
Maemo Compuwere 大数据 aptech Perl Tornado Ruby Hibernate Think→HP HBase
Pure Solr Angular Cloud Foundry Redis Scala Django Bootstrap

公司简介 | 招贤纳士 | 广告服务 | 银行汇款帐号 | 联系方式 | 版权声明 | 法律顾问 | 问题报告 | 合作伙伴 | 论坛反馈

网站客服 杂志客服 微博客服 webmaster@csdn.net 400-600-2320 | 北京创新乐知信息技术有限公司 版权所有 | 江苏乐知网络技术有限公司 提供商务支持

京 ICP 证 070598 号 | Copyright © 1999-2014, CSDN.NET, All Rights Reserved