

DoubleLi

博客园 :: 首页 :: 博问 :: 闪存 :: 新随笔 :: 联系 :: 订阅  :: 管理 ::  2503 随笔 :: 2 文章 :: 232 评论 :: 0 引用

<	2017年2月						>
日	一	二	三	四	五	六	
29	30	31	1	2	3	4	
5	6	7	8	9	10	11	
12	13	14	15	16	17	18	
19	20	21	22	23	24	25	
26	27	28	1	2	3	4	
5	6	7	8	9	10	11	

公告

昵称：[DoubleLi](#)
园龄：7年
粉丝：894
关注：22
[+加关注](#)

搜索

常用链接

[我的随笔](#)
[我的评论](#)
[我的参与](#)
[最新评论](#)
[我的标签](#)

随笔分类(3470)

[ASP.NET\(30\)](#)
[ASP.NET MVC\(11\)](#)
[Boost\(83\)](#)
[c#\(8\)](#)
[C++/C\(612\)](#)
[cmake\(15\)](#)
[com/ATL/Activex\(75\)](#)
[Css\(16\)](#)
[CxImage\(2\)](#)
[DirectX\(16\)](#)
[Extjs\(13\)](#)
[ffmpeg、ffplay\(38\)](#)
[gcc/g++/gdb\(28\)](#)
[h264/h265/mpeg\(3\)](#)
[hisi\(10\)](#)
[html5\(1\)](#)
[http\(5\)](#)
[ICE\(4\)](#)
[java\(13\)](#)
[javascript\(66\)](#)
[jpeglib\(3\)](#)
[Jquery\(9\)](#)
[json\(6\)](#)
[libcurl\(24\)](#)
[Linux\(386\)](#)

Makefile自动生成工具-----autotools的使用（详细）

相信每个学习Linux的人都知道Makefile，这是一个很有用的东西，但是编写它还是比较复杂，今天介绍一个它的自动生成工具，autotools的使用。很多GNU/Linux的软件都是用它生成Makefile的，包括我们非常熟悉的Linux内核源代码。

1、准备：

需要工具

autoscan

aclocal

autoheader

automake

autoconf

auto make

在终端敲入命令，哪个没有安装哪个，一般是第一个autoscan没有，其它的我用的Ubuntu10.04下全部都有

2、测试程序编写：

建立目录：mkdir include src

编写程序：include/str.h

```
1. #include <stdio.h>
2. int str(char *string);
```

编写程序：src/str.c

```
1. #include "str.h"
2. //print string
3. int str(char *string){
4.     printf("\n---PRINT STRING---\n%s\n",string);
5.     return 0;
6. }
7.
8. //interface of this program
9. int main(int argc , char **argv){
10.     char str_read[1024];
11.     printf("Please INPUT something end by [ENTER]\n");
```

linux驱动(18)
live555(18)
makefile(26)
Matlab(3)
mfc control(2)
mingw(4)
mysql(2)
nginx(58)
Nhibernate / Hibernate(3)
OpenCV(23)
OpenGL(1)
Oracle(29)
ProtoBuf(4)
Qt(5)
RTSP/RTP(21)
SDL(8)
shell(3)
SilverLight(1)
sql server(3)
sqlite(5)
SVN/git(27)
tcp/ip(11)
uboot/kernel/rootfs(42)
UML(2)
VC/MFC(579)
web(13)
webserver/CGI(2)
wifi(51)
Windows 编程(30)
Windows控件开发/自绘(17)
WPF(1)
WTL(1)
编解码(12)
操作系统(8)
多线程编程(32)
分布式系统(1)
工程拓扑、视频设备(1)
工具/插件 开发与使用(141)
管理&投资(3)
计算机视觉(10)
计算机网络(4)
架构设计(5)
开源库(36)
其它(6)
嵌入式(198)
数字电路(4)
图像/ISP(2)
图形界面编程(106)
图形学(19)
网络编程(109)
无线网络(38)
异常诊断与调试(104)
音视频、流媒体(141)

随笔档案(2503)

2017年1月 (9)
2016年12月 (38)
2016年11月 (18)
2016年9月 (21)
2016年8月 (10)
2016年7月 (39)
2016年6月 (39)
2016年5月 (26)
2016年4月 (96)

```
12.         scanf("%s",str_read);
13.         return str(str_read );
14.     }
15.
16.
```

3、生成configure.ac

configure.ac是automake的输入文件，所以必须先生成该文件。
执行命令：

```
1. [root@localhost str]# ls
2. include src
3. [root@localhost str]# autoscan
4. autom4te: configure.ac: no such file or directory
5. autoscan: /usr/bin/autom4te failed with exit status: 1
6. [root@localhost str]# ls
7. autoscan.log configure.scan include src
8. [root@localhost str]# cp configure.scan configure.ac
```

修改 configure.ac

```
1. #                                     -*- Autoconf -
2. # Process this file with autoconf to produce a configure scrip
3.
4. AC_PREREQ(2.59)
5. AC_INIT(FULL-PACKAGE-NAME, VERSION, BUG-REPORT-ADDRESS)
6. AC_CONFIG_SRCDIR([include/str.h])
7. AC_CONFIG_HEADER([config.h])
8.
9. # Checks for programs.
10. AC_PROG_CC
11.
12. # Checks for libraries.
13.
14. # Checks for header files.
15.
16. # Checks for typedefs, structures, and compiler characteristic
17.
18. # Checks for library functions.
19. AC_OUTPUT
```

2016年3月 (58)
2016年2月 (4)
2016年1月 (8)
2015年12月 (24)
2015年11月 (23)
2015年10月 (22)
2015年9月 (7)
2015年8月 (32)
2015年7月 (45)
2015年6月 (89)
2015年5月 (74)
2015年4月 (14)
2015年3月 (41)
2015年2月 (3)
2015年1月 (45)
2014年12月 (78)
2014年11月 (20)
2014年10月 (19)
2014年9月 (85)
2014年8月 (52)
2014年7月 (12)
2014年6月 (94)
2014年5月 (105)
2014年4月 (112)
2014年3月 (39)
2014年2月 (40)
2014年1月 (41)
2013年12月 (114)
2013年11月 (141)
2013年10月 (18)
2013年9月 (55)
2013年8月 (69)
2013年7月 (9)
2013年6月 (11)
2013年5月 (55)
2013年4月 (34)
2013年3月 (42)
2013年2月 (12)
2013年1月 (44)
2012年12月 (90)
2012年11月 (60)
2012年10月 (25)
2012年9月 (15)
2012年8月 (46)
2012年7月 (63)
2012年6月 (22)
2012年5月 (12)
2012年4月 (15)
2012年3月 (20)
2012年1月 (10)
2011年12月 (12)
2011年11月 (6)
2011年10月 (14)
2011年9月 (5)
2011年8月 (1)
2011年5月 (1)

文章分类(2)

[SilverLight\(1\)](#)
[sql server\(1\)](#)

参考博客

修改

1. `AC_INIT(FULL-PACKAGE-NAME, VERSION, BUG-REPORT-ADDRESS)`

改为：

1. `AC_INIT(str,0.0.1, [bug@sounos.org])`

其中：FULL-PACKAGE-NAME 为程序名称，VERSION为当前版本，BUG-REPORT-ADDRESS为bug汇报地址

然后添加两句话：

```
AM_INIT_AUTOMAKE
AC_CONFIG_FILES([Makefile])
```

结果如下：（两句话不是在一起的）

```
1. #                                     -*- Autoconf -
2. # Process this file with autoconf to produce a configure scrip
3.
4. AC_PREREQ(2.59)
5. #AC_INIT(FULL-PACKAGE-NAME, VERSION, BUG-REPORT-ADDRESS)
6. AC_INIT(str, 0.0.1, [bug@sounos.org])
7. AM_INIT_AUTOMAKE
8. AC_CONFIG_SRCDIR([include/str.h])
9. AC_CONFIG_HEADER([config.h])
10.
11. # Checks for programs.
12. AC_PROG_CC
13.
14. # Checks for libraries.
15.
16. # Checks for header files.
17.
18. # Checks for typedefs, structures, and compiler characteristic
19.
20. # Checks for library functions.
21. AC_CONFIG_FILES([Makefile])
22. AC_OUTPUT
```

chenyujing1234
com
Dean Chen的专栏
boost 等
linux驱动
morewindows
Nginx模块开发与原理剖析
Nginx模块开发与原理剖析
opencv教程
Sloan
大坡3D软件开发
个人开发历程知识库
关注DirectX
关注DirectX
回忆未来-向东
http://justwinit.cn/index.php
音视频FFmpeg等
音视频FFmpeg等

最新评论

1. Re:非IE内核浏览器支持
activex插件
@Mask-male引用问题已解决，采用URL，而非插件 亲王跟下 您是怎么做的？...
--wuhuan881124
2. Re:非IE内核浏览器支持
activex插件
楼主 能不能加你好友交下我如何做的？
--wuhuan881124
3. Re:hisi出的H264码流结构
还有B帧和那个SPS后面的那个呢。
--Supper_litt
4. Re:非IE内核浏览器支持
activex插件
好东西，楼主 能不能分享一份呀？
543567769@qq.com,谢谢
--爱的angell
5. Re:Nginx之location 匹配规则
详解
对于例2中"但是如果普通 location 的匹配结果恰好是"严格精确 (exact match)"的，则 nginx 不再尝试后面的正则 location "不同意这个观点，不带上"="都还是最.....
--robertch

阅读排行榜

1. Nginx之location 匹配规则详解 (88963)
2. SVN安装配置与使用(76717)
3. C++的Json解析库：jsoncpp和boost .(50922)
4. HTTP 错误 404 - 文件或目录未找到的最终解决方法(36618)
5. C++中的头文件和源文件 (34416)

评论排行榜

4、执行aclocal

1. [root@localhost str]# aclocal
2. /usr/share/aclocal
/libfame.m4:6: warning: underquoted definition of AM_PATH_LIBF
3. run info '(automake)Extending aclocal'
4. or see http://sources.redhat.com/automake
/automake.html#Extending-aclocal

5、制作Makefile.am

1. [root@localhost str]# vi Makefile.am
2. #Makefile.am
3. bin_PROGRAMS = str
4. str_SOURCES = include/str.h src/str.c
5. str_CPPFLAGS = -I include/

automake 这个命令需要用到这个配置文件。各个选项意思比较直观，不多说。

6、autoheader

1. [root@localhost str]# autoheader

7、automake必须文件：

1. * install-sh
2. * missing
3. * INSTALL
4. * NEWS
5. * README
6. * AUTHORS
7. * ChangeLog
8. * COPYING
9. * depcomp

其中，以下文件在执行automake -a的时候会自动生成

1. * install-sh
2. * missing
3. * INSTALL

1. 非IE内核浏览器支持activex插件(20)
2. Javascript中定义类(15)
3. Nginx之location 匹配规则详解(12)
4. C++中的头文件和源文件(8)
5. SVN安装配置与使用(5)

推荐排行榜

1. C++中的头文件和源文件(16)
2. Javascript中定义类(11)
3. Nginx之location 匹配规则详解(11)
4. JavaScript中typeof知多少？(7)
5. SVN安装配置与使用(6)

4. * COPYING
5. * depcomp

所以，接下来手动生成剩下的文件

1. [root@localhost str]# touch NEWS README AUTHORS ChangeLog

8、执行automake -a

1. [root@localhost str]# automake -a
2. configure.ac: installing `./install-sh'
3. configure.ac: installing `./missing'
4. Makefile.am: installing `./INSTALL'
5. Makefile.am: installing `./COPYING'
6. Makefile.am: installing `./compile'
7. Makefile.am: installing `./depcomp'

9、autoconf

1. [root@localhost str]# autoconf
2. [root@localhost str]# ls
3. aclocal.m4 autoscan.log config.h.in configure.scan in
4. AUTHORS ChangeLog configure COPYING IN
5. autom4te.cache compile configure.ac depcomp in

10、执行测试：

执行./configure

1. [root@localhost str]# ./configure --prefix=/u
2. checking for a BSD-compatible install... /usr/bin/install -c
3. checking whether build environment is sane... yes
4. checking for gawk... gawk
5. checking whether make sets \$(MAKE)... yes
6. checking for gcc... gcc
7. checking for C compiler default output file name... a.out
8. checking whether the C compiler works... yes
9. checking whether we are cross compiling... no
10. checking for suffix of executables...
11. checking for suffix of object files... o
12. checking whether we are using the GNU C compiler... yes

```
13. checking whether gcc accepts -g... yes
14. checking for gcc option to accept ANSI C... none needed
15. checking for style of include used by make... GNU
16. checking dependency style of gcc... gcc3
17. configure: creating ./config.status
18. config.status: creating Makefile
19. config.status: creating config.h
20. config.status: config.h is unchanged
21. config.status: executing depfiles commands
```

执行 make

```
1. [root@localhost str]# make
2. make all-am
3. make[1]: Entering directory `/data/devel/c/str'
4. if gcc -DHAVE_CONFIG_H -I. -I. -I. -I include/ -g -O2 -MT s
   /str-str.Tpo" -c -o str-str.o `test -f 'src/str.c' || echo './
   /str.c; \
5. then mv -f ".deps/str-str.Tpo" ".deps
   /str-str.Po"; else rm -f ".deps/str-str.Tpo"; exit 1; fi
6. gcc -g -O2 -o str str-str.o
7. make[1]: Leaving directory `/data/devel/c/str'
```

此时已经生成了 str（可执行文件名字在前面设置Makefile.am的参数时候去顶）
这个，可以通过./str直接看到运行结果

```
1. [root@localhost str]# ./str
2. Please INPUT something end by [ENTER]
3. abcksdhfklsdklfdjlkfd
4.
5. ----PRINT STRING----
6. "abcksdhfklsdklfdjlkfd"
```

不过这里我们都做一步，把它安装到系统里面，这样我们只要在终端输入str就可以运行程序了。

执行 make install：

```
1. [root@localhost str]# make install
2. make[1]: Entering directory `/data/devel/c/str'
3. test -z "/u/bin" || mkdir -p -- "/u/bin"
4. /usr/bin/install -c 'str' '/u/bin/str'
5. make[1]: Nothing to be done for `install-data-am'.
6. make[1]: Leaving directory `/data/devel/c/str'
```

接下来你可以make clean 清除安装的那些.o 文件了。

这样生成了一个自动的Makefile。

分类: [Linux](#)

好文要顶

关注我

收藏该文



DoubleLi

关注 - 22

粉丝 - 894

0

0

+加关注

« 上一篇 : [Linux环境下使用eclipse开发C++动态链接库程序](#)

» 下一篇 : [linux安装Eclipse c++环境](#)

posted on 2014-06-24 13:29 [DoubleLi](#) 阅读(1055) 评论(0) [编辑](#) [收藏](#)

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问网站首页](#)。

最新IT新闻：

- [SpaceX完成了回收火箭的静态点火测试](#)
 - [Windows 10 Cloud可能是微软应对Chrome OS的新招](#)
 - [GoPro无人机Karma召回后再次开卖 恐已难与大疆匹敌](#)
 - [保持开源项目健康运行并减少压力的10件事](#)
 - [今日头条收购美国移动短视频公司Flipagram，未透露收购金额](#)
- » [更多新闻...](#)

最新知识库文章：

- [「代码家」的学习过程和学习经验分享](#)
 - [写给未来的程序媛](#)
 - [高质量的工程代码为什么难写](#)
 - [循序渐进地代码重构](#)
 - [技术的正宗与野路子](#)
- » [更多知识库文章...](#)

历史上的今天：

2013-06-24 [VC工程中的.rc文件和.rc2文件的区别](#)

Powered by:

[博客园](#)

Copyright © DoubleLi