需要达成共识：

ccache是经过严格编写的。它能确保至少在编译结果的正确性上和直接使用make所得到的结果完全一致。甚至不同工程使用相同的ccache也能保证其正确性。

ccache设置

安装ccache

sudo apt-get install ccache

修改.bashrc文件

vi ~/.bashrc

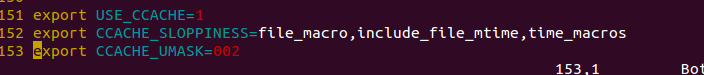
将以下内容添加到末尾

export USE\_CCACHE=1

export CCACHE\_SLOPPINESS=file\_macro,include\_file\_mtime,time\_macros

export CCACHE\_UMASK=002

例如



然后source .bashrc使之在新的终端里生效.

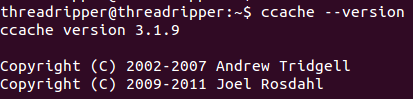
注意:如果之前设置过CCACHE\_DIR,这里需要重新设置

export CCACHE\_DIR=/home/XXXXXX/.ccache XXXXXX是你的用户名

查看ccache版本信息

ccache --version

版本是3.1.9



和工程目录alps/prebuilts/misc/linux-x86/ccache下的ccache文件是同一个版本.

如果不是同一个版本,需要到alps/prebuilts/misc/linux-x86/ccache执行

./ccache -M 100G //设置ccache大小为100G

如果版本相同,任一路径下执行

ccache -M 100G //设置ccache大小为100G

ccache的大小是一个动态的值,每次改变.ccache文件的位置max cache size都会被初始化

查看ccache大小

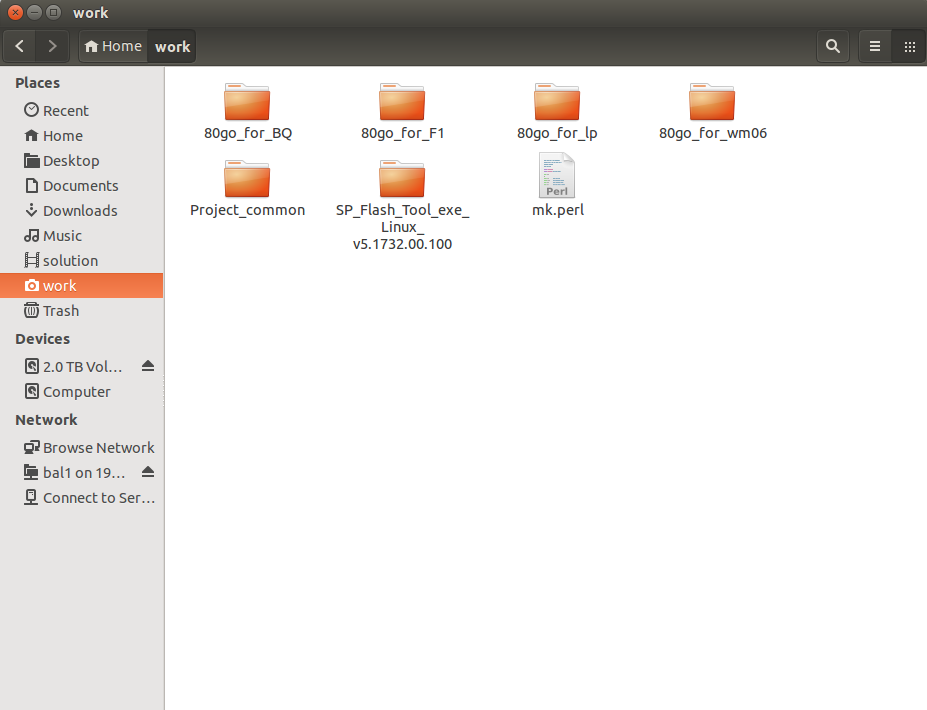
watch -n1 -d ccache -s

创建一个工程目录

这个目录专门用来存放工程文件,如果已经有了就不需要创建.

例如: ~/work

把脚本文件mk.perl放在这个路径下



这些是工程文件



这个是脚本文件



修改权限:

chmod 777 mk.perl

开始编译工程

每次编译工程之前,在工程目录~/work下打开终端

执行./mk.perl 加你要编译的工程文件,文件路径要求是alps和sagereal的上一级目录

例如:

./mk.perl 80go\_for\_BQ 该文件夹下就是alps和sagereal



执行成功后就会提示Project change to XXXXXX success!

第一次执行脚本会创建一个工程文件Project\_common,同时产生一个隐藏文件.project.log,注意不要删除这两个文件,不然会导致mk.perl脚本失效



接下来cd Project\_common 就可以开始编译

下次要编译其他工程只要加上其他工程路径就可以,旧的工程会被放回原来的目录

例如:

执行格80go\_for\_lp式 : ./mk.perl 工程目录

注意:这里要求 工程目录 的下一级目录(子目录)是alps文件和sagereal文件

例如 ./mk.perl 80go\_for\_lp //80go\_for\_lp的下一级是alps和sagereal文件

如果只想把工程放回原路径,需要执行

./mk.perl



Q&A

如何知道自己多个工程在公用一个ccache?

用命令查看ccache

watch -n1 -d ccache -s

如果每次编译都有cache hit (direct)增加就说明已经在公用.

如果我把工程放在不同的磁盘里,可以设置多个工程路径吗?

可以,只需要把mk.perl脚本放到新的工程路径下.

但不建议设置过多工程路径,这样会导致ccache过于臃肿.

ccache加速效果好吗?

最终结果由硬件性能和软件优化决定.

建议使用高速固态硬盘,并且将cache文件放在高速固态硬盘里.

ccache如何清除?

使用ccache -C 清除ccache

使用ccache -z 清除计数

驱动 郑州