

17CY 项目

环境配置

2014/08/05	Init	Kevin Tang
2014/08/22	Update 3-8	Kevin Tang
2014/09/02	modify	liuquan
2014/09/18	Add sdk	liuquan
2014/10/30	Delete ssh KEY	liuquan

目录

第一章 文件服务器访问	3
1.1 Ubuntu 下访问:	3
2.2 Windows 下访问:	3
第二章 操作系统安装	3
第三章 Ubuntu 桌面环境配置	4
3.1 配置公司内网域名解析	4
3.2 配置公司邮箱。	5
3.3 安装开发工具	5
3.4 日常工具安装	8
3.5 一些常用的辅助开发工具	9
3.5.1 图像编辑软件 GIMP	9
3.5.2 JDK 安装	9
3.5.3 安装 Jude(需要先安装 JDK)	10
3.5.4 安装 BeyondCompare	10
3.5.5 安装编辑器	11
3.5.6 安装 Git 及配置	11
3.5.7 安装输入法	12
第四章 基本开发环境配置	12
4.1 配置 SSH	12
4.2 配置 GCC、G++	13
4.3 配置 Repo	14
4.4 下载项目代码	14
第五章 安装正式开发 SDK	15
第六章 MIST 和 QT 相关知识	16

第一章 文件服务器访问

在所有的安装之前，先介绍下如何访问后续安装所需要的文件服务器如何访问。

1.1 Ubuntu 下访问：

ubuntu下, 选择位置->连接到服务器，按照以下参数设置：

服务类型：Windows共享

服务器：172.26.10.69

共享：保留为空

文件夹：保留为空

用户名：保留为空

域名：保留为空

添加书签：选上，书签名可以任意，这方便以后通过“位置”菜单来访问。

进入服务器以后，双击Projects或者References，会提示输入用户名和密码，大家可以使用pubuser的用户，密码为123456。

我们项目的资料地址主要在如下目录：projects/17CY/***

2.2 Windows 下访问：

\\172.26.10.69 即可，用户名为pubuser，密码为123456。如果是Win7用户，用户名可以使用\pubuser，以屏蔽域名。

我们项目的资料地址主要在如下目录：projects/17CY/***

第二章 操作系统安装

我们小组使用的是64位Ubuntu14.04LTS版本. 安装的镜像文件请参考如下地址：
17CY\training\01 - 环境搭建\enviroment\ ubuntu-14.04-desktop-amd64

系统安装前，我只能口头提醒你以下几点：

*** 拔掉网线，防止联网更新**

* 键盘、语言设置等直接下一步

- * 分区选择安装到整个硬盘
- * 机器名设置为姓名全拼

以下是我找的两个网址，仅供参考。

通过上面镜像制作个U盘启动。具体手順可以参考以下的链接。

<http://jingyan.baidu.com/article/59703552e0a6e18fc007409f.html>

通过1的镜像来安装Ubuntu。具体手順可以参考以下的链接。

<http://jingyan.baidu.com/article/dca1fa6fa3b905f1a44052bd.html>

第三章 Ubuntu 桌面环境配置

3.1 配置公司内网域名解析

修改/etc/hosts

```
# sudo vi /etc/hosts
```

红色是追加修正的部分，请黏贴使用。

```
>>>
```

```
127.0.0.1 localhost
```

```
127.0.1.1 tangwenkai
```

```
# The following lines are desirable for IPv6 capable hosts
```

```
::1 ip6-localhost ip6-loopback
```

```
fe00::0 ip6-localnet
```

```
ff00::0 ip6-mcastprefix
```

```
ff02::1 ip6-allnodes
```

```
ff02::2 ip6-allrouters
```

```
172.26.10.41 oa-center
```

```
172.26.10.20 jupiter
```

```
172.26.178.93 igerrit
```

```
172.26.178.153 iwiki
```

```
<<EOF
```

3.2 配置公司邮箱。

14. 04自带了Thunderbird客户端，这个比较好用，请参考以下的文档进行配置。

http://wenku.baidu.com/link?url=nvcFB_0YwdItGdcLkyk8KXqLWYrERaUJ8iLheqodrt7e00GkFqAJeS5azBj-kwhf0YKh0wmPlFXT9geYkuJWxdzNB1Mmbt-NWq8uPELHu1C

3.3 安装开发工具

#64 位系统的32 位软件库支持

```
# sudo apt-get install ia32-libs
```

安装最基本的开发环境

```
# sudo apt-get install bison build-essential libncurses5-dev
```

ccache 安装

```
# sudo apt-get install ccache
```

安装QT支持库

```
sudo apt-get install libqtcore4
```

```
sudo apt-get install libqtgui4
```

```
sudo apt-get install libqt4-network
```

安装必要的软件开发包

```
# sudo apt-get install g++-4.6 gcc-4.6
```

```
# sudo apt-get install g++-4.6-multilib
```

```
# sudo apt-get install g++-4.7-multilib
```

```
# sudo apt-get install android-tools-adb
```

```
# sudo apt-get install libxml2-utils
```

```
# sudo apt-get install libc6-dev
```

```
# sudo apt-get install zlib1g-dev:i386
```

```
# sudo apt-get install libncurses5-dev:i386
```

```
# sudo apt-get install x11proto-core-dev
```

```
# sudo apt-get install libx11-dev:i386
```

```
# sudo apt-get install libreadline6-dev:i386
```

```
# sudo apt-get install libglapi-mesa:i386
```

```
# sudo apt-get install libgl1-mesa-glx:i386
```

```
# sudo apt-get install libgl1-mesa-dev
```

```
# sudo apt-get install libpulse-dev
```

```
# sudo apt-get install libdirectfb-dev
# sudo apt-get install libpangol.0-dev
# sudo apt-get install libdbus-1-dev
# sudo apt-get install libasound2-dev
# sudo apt-get install libesd0-dev
# sudo apt-get install libncurses5-dev
# sudo apt-get install libicu-dev
# sudo apt-get install libxml2-dev
# sudo apt-get install libsqlite3-dev
# sudo apt-get install libegl1-mesa-dev
# sudo apt-get install libgles2-mesa-dev
# sudo apt-get install libevent-dev
# sudo apt-get install libxcursor-dev
# sudo apt-get install libgstreamer0.10-dev
# sudo apt-get install libgstreamer-plugins-base0.10-dev
# sudo apt-get install libtiff4
```

安装必要的软件库

```
# sudo add-apt-repository ppa:mc3man/trusty-media
# sudo apt-get update
# sudo apt-get install gperf
# sudo apt-get install dh-autoreconf
# sudo apt-get install gstreamer0.10-plugins-good
# sudo apt-get install gstreamer0.10-plugins-ugly
# sudo apt-get install gstreamer0.10-plugins-bad
# sudo apt-get install gstreamer0.10-plugins-bad-multiverse
# sudo apt-get install gstreamer0.10-fluendo-mp3
# sudo apt-get install gstreamer0.10-ffmpeg
# sudo apt-get install gstreamer0.10-libav
# sudo apt-get install ffmpeg
# sudo apt-get install pyside-tools
# sudo apt-get install python-m2crypto
# sudo apt-get install python-lxml
# sudo apt-get install lib32z1-dev
# sudo apt-get install g++-multilib
# sudo apt-get install gnupg flex
# sudo apt-get install zip
# sudo apt-get install curl
# sudo apt-get install mingw32
# sudo apt-get install tofrodos
# sudo apt-get install python-markdown
# sudo apt-get install xsltproc
# sudo apt-get install xserver-xorg
# sudo apt-get install xserver-common
```

```
# sudo apt-get install rabbitvcs-nautilus rabbitvcs-gedit
sudo apt-get install libavcodec-dev
sudo apt-get install libavfilter-dev
sudo apt-get install libavformat-dev
sudo apt-get install libavutil-dev
sudo apt-get install libavdevice-dev
sudo apt-get install binutils-dev
sudo apt-get install libc-dev:i386
sudo apt-get install libc6-dev-i386
sudo apt-get install lib32ncurses5-dev
sudo apt-get install libudev-dev
sudo apt-get install liblog4cxx10
sudo apt-get install liblog4cxx10-dev
sudo apt-get install libbluetooth-dev
sudo apt-get install automake
sudo apt-get install automake1.11
sudo apt-get install gstreamer1.0*
sudo apt-get install libqtgstreamer-0.10-0
sudo apt-get install libqtgstreamer-dev
```

```
$ sudo ln -s /usr/lib/i386-linux-gnu/mesa/libGL.so.1
/usr/lib/i386-linux-gnu/libGL.so
```

- * 以上排名不分先后，稍微整理了一下
- * 如遇到互斥软件包，请支一声
- * 如果遇到笔者手误，请先行搜索，再支一声

安装/重装 QtSDK

还未安装Qt的：

QT SDK安装：

- ×将文件服务器上的17CY\training\01 - 环境搭建\enviroment\Qt_opensource-linux-x64-5.3.1.run文件拷贝到本地，比如拷贝到home目录
 - ×打开终端，进入刚才拷贝的目录
 - ×运行./ Qt_opensource-linux-x64-5.3.1.run
- 请注意，SDK要安装到自己的目录下，不要用sudo安装
- 已经安装Qt SDK旧版本的，请先删除旧版本的SDK，然后重新安装
- 1、关闭所有已经打开的QtCreator
 - 2、rm -rf ~/.config/Nokia
 - 3、如果之前使用root权限安装过Qt SDK的，运行sudo rm -rf /opt/QtSDK
 - 4、如果之前使用非root权限安装过Qt SDK的，运行rm -rf ~/QtSDK
 - 5、按照之前的步骤，重装Qt SDK

3.4 日常工具安装

Ubuntu下工具软件众多，日常工作和生活中使用Ubuntu系统已经足够了。下面介绍常用的工具软件的开发。

安装之前，请使用以下命令更新下。

更新软件源

```
# sudo apt-get update
```

更新软件包

```
# sudo apt-get upgrade
```

更新所有软件包

```
# sudo apt-get dist-upgrade
```

* 整个更新过程会比较漫长

在当前位置打开终端

```
sudo apt-get install nautilus-open-terminal
```

安装rar支持

```
sudo apt-get install rar unrar
```

安装字典软件stardict

```
sudo apt-get install stardict
```

装完以后，把字典数据（放在文件服务器的17CY\training\01 - 环境搭建\enviroment\Dictionary\stardict\dic）拷贝到/usr/share/stardict/dic目录

安装ssh支持

```
sudo apt-get install fuse-utils sshfs
```

```
modprobe fuse
```

```
sudo apt-get install ssh
```

安装tftp支持

```
sudo apt-get install tftpd-hpa
```

```
sudo apt-get install tftp
```


安装nfs支持

```
sudo apt-get install nfs-kernel-server
```

安装expect（脚本中经常使用）

```
sudo apt-get install expect
```

安装python支持库：

```
sudo apt-get install python-tk
```

```
sudo apt-get install python-pyside
```

Windows 回车符转换CRLF->LF

```
# sudo apt-get install dos2unix
```

Cmake 工具

```
# sudo apt-get install cmake cmake-gui
```

3.5 一些常用的辅助开发工具

3.5.1 图像编辑软件 GIMP

```
sudo apt-get install gimp
```

3.5.2 JDK 安装

把17CY\training\01 - 环境搭建\enviroment\ jdk-6u37-linux-x64 copy到本地

```
# chmod +x jdk-6u37-linux-x64.bin
```

```
# ./jdk-6u37-linux-x64.bin
```

```
# sudo mkdir -p /usr/lib/jvm/
```

```
# sudo mv jdk1.6.0_37/ /usr/lib/jvm/
```

```
# sudo update-alternatives --install "/usr/bin/java" "java"
```

```
"/usr/lib/jvm/jdk1.6.0_37/bin/java" 1000
```

```
# sudo update-alternatives --config java
```

如图所示：

```
tangwenkai@tangwenkai:~$ sudo update-alternatives --config java
有 2 个候选项可用于替换 java (提供 /usr/bin/java)。
```

选择	路径	优先级	状态
0	/usr/lib/jvm/java-7-openjdk-amd64/jre/bin/java	1071	自动模式
1	/usr/lib/jvm/java-7-openjdk-amd64/jre/bin/java	1071	手动模式
* 2	/usr/lib/jvm/jdk1.6.0_37/bin/java	1000	手动模式

要维持当前值[*]请按回车键，或者键入选择的编号：2

```
# update-alternatives --install "/usr/bin/javac" "javac"
"/usr/lib/jvm/jdk1.6.0_37/bin/javac" 1000
# sudo update-alternatives --config javac
如图所示：
```

```
tangwenkai@tangwenkai:~$ sudo update-alternatives --config javac
有 2 个候选项可用于替换 javac (提供 /usr/bin/javac)。
```

选择	路径	优先级	状态
0	/usr/lib/jvm/java-7-openjdk-amd64/bin/javac	1071	自动模式
1	/usr/lib/jvm/java-7-openjdk-amd64/bin/javac	1071	手动模式
* 2	/usr/lib/jvm/jdk1.6.0_37/bin/javac	1000	手动模式

要维持当前值[*]请按回车键，或者键入选择的编号：

3.5.3 安装 Jude(需要先安装 JDK)

将文件服务器上的17CY\training\01 - 环境搭建
 \enviroment\astah-community_6.5-1_all.deb拷贝到本地，在ubuntu下找到本地文件，打开终端，运行

```
sudo dpkg -i astah-community_6.5-1_all.deb
```

注意安装完成之后，会提示失败，无视就好。

使用软件的时候，会提示更新，请不要更新，就用这个版本。

3.5.4 安装 BeyondCompare

BeyondCompare是一款很优秀的比较软件，比较幸运的是，BC现在有了官方的Ubuntu版本，我已经下来了一份放在17CY\training\01 - 环境搭建
 \enviroment\bcompare-3.3.3.14128_i386.deb。把这个文件拷到本地，双击即可安装。

这个版本是30天试用版，如果30天将要到期了，可以删除他的配置文件，然后又
 可以继续用了。

```
rm -rf ~/.beyondcompare
```

3.5.5 安装编辑器

```
# sudo apt-get install vim
```

3.5.6 安装 Git 及配置

安装:

```
sudo apt-get install git git-core  
sudo apt-get install gitk
```

配置:

```
# git config --global user.name xxx (注: xxx 为申请的邮箱用户名)  
# git config --global user.email xxx@pset.suntec.net (注: xxx 为申请的  
邮箱用户名)  
# git config --global core.autocrlf input (注: 行结束符在签入时转换为  
LF, 签出时  
不变)  
# git config --global core.safecrlf true (注: 签入时检查行结束符, 不  
允许混合)  
# git config --global color.ui true (注: 打开git色彩显示)
```

配置Bycompare为外部Diff工具。

1. vi ~/.gitconfig
2. [diff]
Tool = bc3
[difftool]
Prompt = false
[merge]
Tool = bc3
[mergetool]
Prompt =false

配置完成。

再配置GUI端:

需要进入代码仓库。

Cmd 输入 gitk

选择Edit-> Preferences -> External Tool 选择 /usr/lib/bcompare

3. Diff 文件:

Git difftool xxx文件

或者在git GUI界面上选择文件之后, 选择外部diff工具, 就可以用BC了。

3.5.7 安装输入法

```
# sudo apt-get install fcitx-googlepinyin
```

安装语言包，或许中文会比较友好

打开应用程序->系统工具->系统设置->语言支持

点击添加删除语言，勾选中文，然后应用变更

* 在列表中，把中文拖到最前面

* 键盘输入方式选择fcitx，当然ibus 也是可以的

至此，重启以后就是中文界面了

* 不要把桌面、文档、下载等文件夹更换成中文目录，否则命令行操作会很麻烦

第四章 基本开发环境配置

4.1 配置 SSH

1. 生成Key:

```
# ssh-keygen -t rsa #生成密钥文件
```

```
# cd ~/.ssh #你的私钥为id_rsa, 公钥为id_rsa.pub
```

```
# cp id_rsa.pub YOUR_NAME.pub #拷贝便于识别
```

然后把你的公钥copy到如下的位置: 17CY\members\公钥\申请中
之后17CY环境搭建成功，会帮助大家申请权限。

2. 登陆Key:

```
# ssh-add id_rsa #将私钥添加到认证代理中打开浏览器登录
```

<http://igerrit.storm>

进入Setting->SSH Public Key->Add Key->复制你的公钥然后点击Add

打开浏览器登录<http://review.iauto.net/iauto/>

进入Setting->SSH Public Key->Add Key->复制你的公钥然后点击Add

* 注意备份你的私钥和公钥，以防系统出现问题

注意，添加的时候，公钥后面的机器名或者邮件地址就不要copy上去了。

3. 追加Config:

在本地的~/.ssh文件夹下新建config文件

将以下内容copy到config文件，并将 User 替换成自己的storm账号。

```
Host iauto
  Hostname review.iauto.net
  IdentityFile ~/.ssh/id_rsa
  User iautoci
  Port 29418
```

```
Host igerrit.storm
  Hostname igerrit.storm
  IdentityFile ~/.ssh/id_rsa
  User iautoci
```

4. 测试确认:

在命令行中输入: ~~ssh -p 29418 iauto~~ 如果验证成功会有如下提示:

~~**** Welcome to Gerrit Code Review ****~~

~~Hi iAutoslave, you have successfully connected over SSH.~~

~~Unfortunately, interactive shells are disabled. To clone a hosted Git repository, use:~~

~~git clone ssh://iautoslave@review.iauto.net:29418/REPOSITORY_NAME.git~~

~~Connection to review.iauto.net closed.~~

4.2 配置 GCC、G++

```
# sudo update-alternatives --install /usr/bin/gcc gcc /usr/bin/gcc-4.7
40
```

```
# sudo update-alternatives --install /usr/bin/gcc gcc /usr/bin/gcc-4.6
30
```

```
# sudo update-alternatives --config gcc
```

如图所示:

```
tangwenkai@tangwenkai:~$ sudo update-alternatives --config gcc
有 2 个候选项可用于替换 gcc (提供 /usr/bin/gcc)。
```

选择	路径	优先级	状态
0	/usr/bin/gcc-4.7	40	自动模式
* 1	/usr/bin/gcc-4.6	30	手动模式
2	/usr/bin/gcc-4.7	40	手动模式

要维持当前值[*]请按回车键，或者键入选择的编号：1

```
# sudo update-alternatives --install /usr/bin/g++ g++ /usr/bin/g++-4.7
```

```
100
# sudo update-alternatives --install /usr/bin/g++ g++ /usr/bin/g++-4.6
80
# sudo update-alternatives --config g++
```

如图所示：

```
tangwenkai@tangwenkai:~$ sudo update-alternatives --config g++
有 2 个候选项可用于替换 g++ (提供 /usr/bin/g++)。
```

选择	路径	优先级	状态
0	/usr/bin/g++-4.7	100	自动模式
* 1	/usr/bin/g++-4.6	80	手动模式
2	/usr/bin/g++-4.7	100	手动模式

```
要维持当前值 [*] 请按回车键，或者键入选择的编号：1
```

4.3 配置 Repo

至此，你的公钥应该上传至服务器了，然后你就可以拉取代码了

* 如果出现提示要求输入密码，或者没有权限之类，请通知xujuan追加权限。

```
git clone ssh://igerrit.storm:29418/tools/repo.git
cd repo
git checkout -b stable origin/stable
sudo cp ./repo /usr/bin/repo
```

删除本地repo仓库

```
# sudo rm -rf ../repo
```

4.4 下载项目代码

1. 建立工作目录

```
mkdir 17Cy
```

```
cd 17Cy
```

2. 取得代码仓库清单

```
repo init -u ssh://igerrit.storm:29418/Src/17Model/17Cy/manifest -b
master
```

常见问题

如果取代码时出现无法解析主机名的错误：

```
ssh: Could not resolve hostname igerrit: Name or service not known
fatal: The remote end hung up unexpected
```

配置域名访问就可以了

3. 同步代码

```
repo sync
```

4. 切换分支

```
repo start master --all
```

5. 以后有修正提交。

```
git push origin master:refs/for/master
```



* Simulator 和Emulator 的差异可以百度，个人理解前者是纯软件模拟，不涉及

到内核设备的模拟等，后者仿真度高一点，更接近实机

* 如果编译过程中提示缺少依赖库的话，用之前的命令进行搜索吧

* 以上命令如有手误，敬请谅解，也欢迎补充

第五章 安装正式开发 SDK

//172.26.11.229/release/iAuto/Release (这个是1.0的sdk)

1.0 请参考SDK Manual for Qt5.odt进行环境配置

//172.26.11.229/release/iAuto/iAuto2.0_release/sdk/renesas_sdk/ (这个是2.0的sdk,正式项目开发用)

2.0的请参考上面目录下的SDK Manual for Iauto2.0(20141029).pdf来安装。

第六章 MIST 和 QT 相关知识

请注意：MIST 主要是用来画 UI 状态迁移图的，由于在项目中，基本上都是 Leader 或者专员来画。所以画面担当的话，只需要学会看图，知道从这个画面迁移到另外一个画面，该发送哪个 Trigger 或者设置哪个 choice 即可。

QT 相关的知识，请要熟悉并了解，因为跟开发息息相关。

MIST 下载地址：

MIST 的 WINDOWS 版本比 Linux 版本稳定性要好些。

安装包：<ftp://cmstest3.pset.suntec.net/MIST/spec2.0/pre-release-merge/>

升级包地址：ftp://cmstest3.pset.suntec.net/MIST/spec2.0/update_linux/

Mist 和 QT 的介绍

http://cmstest3.pset.suntec.net/sdkdoc/post13/#!/guide/qt_guide_contents