

Deepin LiveCD 构建流程

张成

构建 Live ISO 的基本步骤

- 使用 debootstrap 构建基本 Ubuntu 的基本系统
- **chroot** 到 Ubuntu 的基本系统中，安装 deepin 预装软件
 - 安装 deepin-meta-desktop 以及其他没有被这个包依赖的 deepin 预装软件
 - 安装 casper, 这个工具是 LiveCD 的核心，有了它 LiveCD 才能启动
 - 安装 ubiquity, 这个工具是系统的安装器
- 对定制好的系统进行清理工作，删除临时文件、缓存文件等
 - 清理 /var/cache 中无用的东西
 - 其他定制工作
 - 清理 /root/.bash_history
- 将系统压缩打包成 squashfs
- 制作 ISO

脚本的使用

- 脚本的 git 仓库临时存放在 mirrors:/git-repos/中：
`git clone mirrors:/git-repos/deepin-system`

- 脚本的使用方法：

- 在系统中安装构建 iso 所需要的工具，一台机器只需执行一次

```
./mkdeepin init
```

- 构建 iso 的四个步骤可以一次执行

```
./mkdeepin -arch=i386 base custom casper iso
```

配置文件的结构

```
/-  
  |- mkdeepin  
  |- mkdeepin.conf  
  |- mkdeepin.conf.local  
  |- conf/  
    |- 11.06  
    |- 11.12  
    |- current -> 11.12  
      |- desktop  
      |- 其他实验版  
      |- default -> desktop  
        |- precustom.{all,i386,amd64}.sh{,.local}  
        |- precustom.chroot.{all,i386,amd64}.sh{,.local}  
        |- postcustom.{all,i386,amd64}.sh{,.local}  
        |- postcustom.chroot.{all,i386,amd64}.sh{,.local}  
        |- preiso.{all,i386,amd64}.sh{,.local}  
        |- ...
```

构建 ISO 耗时

第一步, 构建基本系统, 约占 10~15%

第二步, 定制系统, 约占50~60%

第三步, 打包 squashfs, 约占30%

第四步, 生成 ISO, 约占10%

- 在工作机器上, 用磁盘构建 ISO, 约23分钟, 用内存构建 ISO, 约6分钟。
- 在老服务器上, 用磁盘构建 ISO, 约40分钟, 内存较小, 不适合在内存构建 ISO。
- 在Google服务器上, 用内存构建约19分钟

构建时机

- 修改定制脚本之后
- 每日常规更新，更新仓库中最新的软件

脚本优化

- 构建基本系统
 - 每周使用同一个基本系统，减少安装基本系统的时间
- 定制系统
 - 进一步系统定制脚本，分离出可重入的代码，减少重复安装预装软件的时间
- 打包 filesystem.squashfs
 - 暂时没有优化的方案
- 生成 ISO
 - 生成 ISO 的时间本生很快，但如果仅仅为了在 PXE 上测试，则不需要生成 ISO

脚本完成进度

公网访问

```
cdimage.linuxdeepin.com /  
  |- releases/  
    |- 11.06/  
    |- 11.12/  
    |- ...  
  |- daily-live/  
    |- 20111021/  
    |- 20111022/  
    |- 20111023/  
    |- current -> 20111023  
  |- deepin-gnome3/  
  |- deepin-ooxx/
```

公网分发方式

由于可能会带来较大的流量，因此考虑只允许使用zsync下载，在服务器上配置nginx，拒绝非zsync的User Agent。