

Debian Installer Internals

Frans Pop

DebConf 6, Oaxtepec,
Mexico



- 1 Introduction
- 2 Installation methods
- 3 Running the installer
- 4 Debugging the installer
- 5 Creating udebs
- 6 Building installer images



Outline

- 1 Introduction
- 2 Installation methods
- 3 Running the installer
- 4 Debugging the installer
- 5 Creating udebs
- 6 Building installer images



So what are we going to do?

- Installation methods
- What happens when the installer is booted and running?
- Debug the installer
- Create a udeb
- Build an installer image



Some resources

- `<debian-boot@lists.debian.org>`
- <http://www.debian.org/devel/debian-installer>
- <http://wiki.debian.org/DebianInstaller>
- <http://d-i.alioth.debian.org/manual/>
- `$ svn co svn+ssh://svn.debian.org/svn/d-i/trunk`



Outline

- 1 Introduction
- 2 Installation methods**
- 3 Running the installer
- 4 Debugging the installer
- 5 Creating udebs
- 6 Building installer images



Basic stages

- 1 boot and initialization
- 2 loading additional components
- 3 network configuration (unless already done in 1st stage)
- 4 partitioning
- 5 installing the target system



Stages 1 - 3

Stage	CD-ROM	NETBOOT
-		initrd-preseed
1		localechooser
1		kbd-chooser
1	cdrom-detect	eth-detect
1		netcfg
-	file-preseed	network-preseed
2		choose-mirror
2	load-cdrom (anna)	download-installer (anna)
3	eth-detect	
3	netcfg	
3	choose-mirror	



Stages 4 & 5

Stage	All methods
4	hw-detect
4	partman
5	tzsetup
5	clock-setup
5	user-setup
5	base-installer
5	apt-setup
5	pkgsel
5	grub/lilo-installer; nobootloader
5	finish-install (was: prebaseconfig)



Installation methods

Installation method is determined by 4 characteristics

- how is the installer booted
- from where are additional udebs retrieved
- from where are packages for the base system retrieved
- from where are packages for tasks retrieved



Most common installation methods

Method	Boot	Udebs	Base system	Tasks
netboot	network (TFTP server)	network	network	network
mini.iso	CD	network	network	network
businesscard CD	CD	CD	network	network
netinst CD	CD	CD	CD	network
full CD/DVD	CD	CD	CD	CD (+ network)
hd-media	harddisk/USB stick	CD image	CD image/network	CD image/network
floppy (net)	boot/root/net-drivers	network	network	network
floppy (cd)	boot/root/cd-drivers	CD	CD/network	CD/network



Outline

- 1 Introduction
- 2 Installation methods
- 3 Running the installer**
- 4 Debugging the installer
- 5 Creating udebs
- 6 Building installer images



Boot process

- **/init** (initramfs) or **/sbin/init** (initrd)
- **busybox init** (parses /etc/inittab)
 - ::sysinit:/sbin/debian-installer-startup
 - ::respawn:/sbin/debian-installer
 - init for VT2 (busybox shell), VT3 (/var/log/messages), VT4 (/var/log/syslog)
- **/sbin/debian-installer-startup**
runparts on /lib/debian-installer-startup.d
- **/sbin/debian-installer**
runparts on /lib/debian-installer.d; starts main menu

Scripts in /lib/debian-installer(-startup).d can be architecture-specific.



Main menu

- not visible at critical and high debconf priority
- priority adjusted automatically on error and <Go Back>
- dynamically assembled
- order determined by dependencies and menu numbers
- selection of menu item executes postinst
- items with menu number higher than finish-install are not run automatically



Menu udebs in dpkg/status file

```
Package: netcfg  
Status: install ok installed  
Version: 1.23  
Provides: configured-network  
Depends: libc6 (>= 2.3.5-1), libdebconfclient0, libdebian-installer4 (>= 0.37),  
        dhcp-client-udeb | dhcp3-client-udeb | pump-udeb, libiw28-udeb,  
        cdebconf-udeb, ethernet-card-detection  
Description: Configure the network  
Installer-Menu-Item: 18
```

```
Package: choose-mirror  
Status: install ok unpacked  
Version: 1.19  
Depends: libc6 (>= 2.3.5-1), libdebconfclient0, libdebian-installer4 (>= 0.38),  
        configured-network  
Description: Choose mirror to install from  
Installer-Menu-Item: 23
```



Hooks

Hooks are used to provide additional flexibility

- `/usr/lib/base-installer.d`
Run by base-installer before debootstrap is started
- `/usr/lib/post-base-installer.d`
Run by base-installer just before kernel selection/installation
- `/usr/lib/finish-install.d` Run at the end of the installation

Other hooks

- `/lib main-menu.d`
- `/usr/lib/apt-setup/generators`
- `/lib/rescue.d`
- `partman`



Special tools

- `anna-install`
- `apt-install`
- `log-output`
- `in-target`



Preseeding

See the paper...



Outline

- 1 Introduction
- 2 Installation methods
- 3 Running the installer
- 4 Debugging the installer**
- 5 Creating udebs
- 6 Building installer images



Debugging

Let's just do it!



Outline

- 1 Introduction
- 2 Installation methods
- 3 Running the installer
- 4 Debugging the installer
- 5 Creating udebs**
- 6 Building installer images



Creating udebs

- optimized for size
- relaxed policy requirements for binary packages
- no version management
- fully supported by debhelper
- types of udebs
 - rootskel
 - kernel image and kernel module udebs
 - menu items and components
 - library and utility udebs



udeb dependencies

Recently added "type" support in shlibs files

- `libz 1 zlib1g (>= 1:1.2.1)`
`udeb: libz 1 zlib1g-udeb (>= 1:1.2.1)`
- `dh_makeshlibs -V -s --add-udeb="libusb-0.1-udeb"`



debian/control

```
Source: kbd-chooser
Section: debian-installer
Priority: optional
Maintainer: Debian Install System Team <debian-boot@lists.debian.org>
Uploaders: [...]
Build-Depends: debhelper (>= 5.0.22), libdebian-installer4-dev (>= 0.41),
               po-debconf (>= 0.5.0), flex | flex-old , bison,
               libdebconfclient0-dev (>= 0.49)
```

```
Package: kbd-chooser
Architecture: i386 amd64 powerpc alpha hppa sparc [...]
XC-Package-Type: udeb
Depends: ${shlibs:Depends}, ${misc:Depends}, console-keymaps
Description: Detect a keyboard and select layout
XB-Installer-Menu-Item: 12
```



Let's create a udeb

What is the one killer feature that most installers have,
but that Debian Installer lacks?



Let's create a udeb

Entry of a licence key

Credits for this idea go to Martin Zobel-Helas



Outline

- 1 Introduction
- 2 Installation methods
- 3 Running the installer
- 4 Debugging the installer
- 5 Creating udebs
- 6 Building installer images**



Images

- Most d-i images are "ready for use"
- Exception: d-i images for CD-ROMs => debian-cd
- Exception to the exception: mini.iso
- Images can be built separately or as part of a release



Release

- dpkg-buildpackage
- upload
- BYHAND
- builddds
- more BYHAND



Requirements for building

To build the current development version:

- unstable environment
- SVN checkout of trunk
- apt-get build-dep debian-installer
- available mirror
 - sources.list.udeb: based on /etc/apt/sources.list
 - sources.list.udeb.local: manual

To build the Sarge installer a stable environment and a checkout of the sarge branch are needed.



Structure of the build system

- based on 'make'
- hierarchical and recursive
- config: defines available targets (per architecture)
- pkg-lists: defines which udebs go into an image (per type)
- boot: configuration files and make targets to make images bootable
- localudebs: allows to include debs not on the mirror
- util: contains helper scripts called from the Makefile
- tmp: working directory where image is "assembled"
- dest: built images, logfiles and manifests



Library reduction

- Used to minimize the size of the initrd
- Selected libraries are relinked, leaving out unused symbols
- For some libs the full version is installed during the installation
- Sometimes the cause of trouble...
- See paper for further info

