

# Custom Debian Distributions at starting block

## Debian-Edu as pacemaker

*Andreas Tille*

Nafplion, April 2005

# Structure

1

## Ready

- History
- CDD Tool
- Differences

2

## Steady

- CDD tools rewrite
- Web page generation

3

## Go



# Structure

1

## Ready

- History
- CDD Tool
- Differences

2

## Steady

- CDD tools rewrite
- Web page generation

3

## Go



# Structure

1

## Ready

- History
- CDD Tool
- Differences

2

## Steady

- CDD tools rewrite
- Web page generation

3

## Go



1

## Ready

- History

- CDD Tool

- Differences

2

## Steady

- CDD tools rewrite

- Web page generation

3

## Go



# Debian based school distributions

- Debian-Edu (became MIA)
- SkoleLinux (very active)
- Since DebConf 3 in Oslo SkoleLinux became Debian-Edu
- School distributions in Spain
- ...



# Debian based school distributions

- Debian-Edu (became MIA)
- SkoleLinux (very active)
- Since DebConf 3 in Oslo SkoleLinux became Debian-Edu
- School distributions in Spain
- ...



# Debian based school distributions

- Debian-Edu (became MIA)
- SkoleLinux (very active)
- Since DebConf 3 in Oslo SkoleLinux became Debian-Edu
- School distributions in Spain
- ...





# Debian based school distributions

- Debian-Edu (became MIA)
- SkoleLinux (very active)
- Since DebConf 3 in Oslo SkoleLinux became Debian-Edu
- School distributions in Spain
- ...



# Debian based school distributions

- Debian-Edu (became MIA)
- SkoleLinux (very active)
- Since DebConf 3 in Oslo SkoleLinux became Debian-Edu
- School distributions in Spain
- ...



# Techniques we owe SkoleLinux

- ***Installer !!!***

What would be Debian without it?

- Building meta-packages regarding different distributions
- Enables building meta packages
- Pre-seeding *debconf*
- ...



# Techniques we owe SkoleLinux

- *Installer !!!*

What would be Debian without it?

- Building meta-packages regarding different distributions

- Enables building meta packages

- Pre-seeding *debconf*

- ...



# Techniques we owe SkoleLinux

- *Installer !!!*

What would be Debian without it?

- Building meta-packages regarding different distributions
- Enables building meta packages
- Pre-seeding *debconf*
- ...



# Techniques we owe SkoleLinux

- *Installer !!!*

What would be Debian without it?

- Building meta-packages regarding different distributions
- Enables building meta packages
- Pre-seeding *debconf*

● . . .



# Techniques we owe SkoleLinux

- *Installer !!!*

What would be Debian without it?

- Building meta-packages regarding different distributions
- Enables building meta packages
- Pre-seeding *debconf*
- ...



1

## Ready

- History
- **CDD Tool**
- Differences

2

## Steady

- CDD tools rewrite
- Web page generation

3

## Go





# Building meta-packages

- *Cdd-dev* respects Debian-Edu layout
- Creates *tasksel* file but more general
- New feature: user menus



# Building meta-packages

- *Cdd-dev* respects Debian-Edu layout
- Creates *tasksel* file but more general
- New feature: user menus



# Building meta-packages

- *Cdd-dev* respects Debian-Edu layout
- Creates *tasksel* file but more general
- New feature: user menus



**My offer:**

**My offer:**

**Porting meta-package  
building stuff to new  
system**

1

## Ready

- History
- CDD Tool
- Differences

2

## Steady

- CDD tools rewrite
- Web page generation

3

## Go



# Common Meta-package

## Optional

```
$ cp -a /usr/share/doc/cdd-dev/examples/common .  
$ cat common/README  
$ edit common/{conf,control,common.1}  
change variable _CDD_ to edu
```

*Initialisation of role based menu system*

*Installation of the role based menu system*

*Meta packages will depend from common package*

*is different from edu-cdd-to-common*



# Common Meta-package

## Optional

```
$ cp -a /usr/share/doc/cdd-dev/examples/common .  
$ cat common/README  
$ edit common/{conf,control,common.1}  
change variable _CDD_ to edu
```

## *Initialisation of role based menu system*

- Meta packages will depend from common package
- Different from `educdd-to-common`





# Common Meta-package

## Optional

```
$ cp -a /usr/share/doc/cdd-dev/examples/common .  
$ cat common/README  
$ edit common/{conf,control,common.1}  
change variable _CDD_ to edu
```

## *Initialisation of role based menu system*

- Meta packages will depend from common package
- Different from education-common



# Common Meta-package

## Optional

```
$ cp -a /usr/share/doc/cdd-dev/examples/common .  
$ cat common/README  
$ edit common/{conf,control,common.1}  
change variable _CDD_ to edu
```

## *Initialisation of role based menu system*

- Meta packages will depend from common package
- Different from education-common



# Common Meta-package

## Optional

```
$ cp -a /usr/share/doc/cdd-dev/examples/common .  
$ cat common/README  
$ edit common/{conf,control,common.1}  
change variable _CDD_ to edu
```

## *Initialisation of role based menu system*

- Meta packages will depend from common package
- Different from education-common



# Common Meta-package

## Optional

```
$ cp -a /usr/share/doc/cdd-dev/examples/common .  
$ cat common/README  
$ edit common/{conf,control,common.1}  
change variable _CDD_ to edu
```

## *Initialisation of role based menu system*

- Meta packages will depend from common package
- Different from education-common



# Common Meta-package

## Optional

```
$ cp -a /usr/share/doc/cdd-dev/examples/common .  
$ cat common/README  
$ edit common/{conf,control,common.1}  
change variable _CDD_ to edu
```

## *Initialisation of role based menu system*

- Meta packages will depend from common package
- Different from education-common



# Common Meta-package

## Optional

```
$ cp -a /usr/share/doc/cdd-dev/examples/common .  
$ cat common/README  
$ edit common/{conf,control,common.1}  
change variable _CDD_ to edu
```

## *Initialisation of role based menu system*

- Meta packages will depend from common package
- Different from education-common



# Role based user menus

- User with role `<cdd>` gets extra user menu



- User with role `<cdd>` is provided with additional menu
- Do not bother other users with those additional menus
- Implemented in current Debian-Med packages
- Role system is implemented by flexible plugins



# Role based user menus

- User with role `<cdd>` gets extra user menu



- User with role `<cdd>` is provided with additional menu
- Do not bother other users with those additional menus
- Implemented in current Debian-Med packages
- Role system is implemented by flexible plugins





# Role based user menus

- User with role `<cdd>` gets extra user menu



- User with role `<cdd>` is provided with additional menu
- Do not bother other users with those additional menus
- Implemented in current Debian-Med packages
- Role system is implemented by flexible plugins



# Role based user menus

- User with role `<cdd>` gets extra user menu



- User with role `<cdd>` is provided with additional menu
- Do not bother other users with those additional menus
- Implemented in current Debian-Med packages
- Role system is implemented by flexible plugins



# Role based user menus

- User with role `<cdd>` gets extra user menu



- User with role `<cdd>` is provided with additional menu
- Do not bother other users with those additional menus
- Implemented in current Debian-Med packages
- Role system is implemented by flexible plugins



# Changing user menus

## Optional

```
$ cp -a /usr/share/doc/cdd-dev/examples/menu .  
$ cat menu/README  
$ cat menu/task1/README  
$ edit menu/task1/dep1.menu  
$ edit menu/task1/dep2.txt  
$ edit menu/task1/dep3.html
```

Create valid menu entries with sensible descriptions

Move to right task menu

```
$ cp menu/task1/menu/cdd-dev-perkins menu/task2
```

*User menu entries for every dependencies - at least pointing to documentation*



# Changing user menus

## Optional

```
$ cp -a /usr/share/doc/cdd-dev/examples/menu .  
$ cat menu/README  
$ cat menu/task1/README  
$ edit menu/task1/dep1.menu  
$ edit menu/task1/dep2.txt  
$ edit menu/task1/dep3.html
```

Create valid menu entries with sensible descriptions

Move to right task name

```
$ cp menu/task1 menu/<meta package name>
```

*User menu entries for every dependencies - at least  
pointing to documentation*



# Changing user menus

## Optional

```
$ cp -a /usr/share/doc/cdd-dev/examples/menu .  
$ cat menu/README  
$ cat menu/task1/README  
$ edit menu/task1/dep1.menu  
$ edit menu/task1/dep2.txt  
$ edit menu/task1/dep3.html
```

Create valid menu entries with sensible descriptions

Move to right task name

```
$ cp menu/task1 menu/<meta package name>
```

*User menu entries for every dependencies - at least pointing to documentation*



# Changing user menus

## Optional

```
$ cp -a /usr/share/doc/cdd-dev/examples/menu .  
$ cat menu/README  
$ cat menu/task1/README  
$ edit menu/task1/dep1.menu  
$ edit menu/task1/dep2.txt  
$ edit menu/task1/dep3.html
```

Create valid menu entries with sensible descriptions

Move to right task name

```
$ cp menu/task1 menu/<meta package name>
```

*User menu entries for every dependencies - at least pointing to documentation*



# Changing user menus

## Optional

```
$ cp -a /usr/share/doc/cdd-dev/examples/menu .  
$ cat menu/README  
$ cat menu/task1/README  
$ edit menu/task1/dep1.menu  
$ edit menu/task1/dep2.txt  
$ edit menu/task1/dep3.html
```

Create valid menu entries with sensible descriptions

Move to right task name

```
$ cp menu/task1 menu/<meta package name>
```

*User menu entries for every dependencies - at least pointing to documentation*





# Changing user menus

## Optional

```
$ cp -a /usr/share/doc/cdd-dev/examples/menu .  
$ cat menu/README  
$ cat menu/task1/README  
$ edit menu/task1/dep1.menu  
$ edit menu/task1/dep2.txt  
$ edit menu/task1/dep3.html
```

Create valid menu entries with sensible descriptions

Move to right task name

```
$ cp menu/task1 menu/<meta package name>
```

*User menu entries for every dependencies - at least pointing to documentation*



# Changing user menus

## Optional

```
$ cp -a /usr/share/doc/cdd-dev/examples/menu .  
$ cat menu/README  
$ cat menu/task1/README  
$ edit menu/task1/dep1.menu  
$ edit menu/task1/dep2.txt  
$ edit menu/task1/dep3.html
```

Create valid menu entries with sensible descriptions

Move to right task name

```
$ cp menu/task1 menu/<meta package name>
```

*User menu entries for every dependencies - at least pointing to documentation*



# Changing user menus

## Optional

```
$ cp -a /usr/share/doc/cdd-dev/examples/menu .  
$ cat menu/README  
$ cat menu/task1/README  
$ edit menu/task1/dep1.menu  
$ edit menu/task1/dep2.txt  
$ edit menu/task1/dep3.html
```

Create valid menu entries with sensible descriptions

Move to right task name

```
$ cp menu/task1 menu/<meta package name>
```

*User menu entries for every dependencies - at least pointing to documentation*



# Changing user menus

## Optional

```
$ cp -a /usr/share/doc/cdd-dev/examples/menu .  
$ cat menu/README  
$ cat menu/task1/README  
$ edit menu/task1/dep1.menu  
$ edit menu/task1/dep2.txt  
$ edit menu/task1/dep3.html
```

Create valid menu entries with sensible descriptions

## Move to right task name

```
$ cp menu/task1 menu/<meta package name>
```

*User menu entries for every dependencies - at least pointing to documentation*



# Changing user menus

## Optional

```
$ cp -a /usr/share/doc/cdd-dev/examples/menu .  
$ cat menu/README  
$ cat menu/task1/README  
$ edit menu/task1/dep1.menu  
$ edit menu/task1/dep2.txt  
$ edit menu/task1/dep3.html
```

Create valid menu entries with sensible descriptions

## Move to right task name

```
$ cp menu/task1 menu/<meta package name>
```

*User menu entries for every dependencies - at least pointing to documentation*



- 1 Ready
  - History
  - CDD Tool
  - Differences
- 2 **Steady**
  - **CDD tools rewrite**
  - Web page generation
- 3 Go



# User menus

- Find out what is missing in Debian's *menu* package
- Fix Debian's *menu* (if necessary)
- Add feature of user groups to *cdd-tool*
- Start using common menu techniques in SkoleLinux



# User menus

- Find out what is missing in Debian's *menu* package
- Fix Debian's *menu* (if necessary)
- Add feature of user groups to *cdd-tool*
- Start using common menu techniques in SkoleLinux





# User menus

- Find out what is missing in Debian's *menu* package
- Fix Debian's *menu* (if necessary)
- Add feature of user groups to *cdd-tool*
- Start using common menu techniques in SkoleLinux



# User menus

- Find out what is missing in Debian's *menu* package
- Fix Debian's *menu* (if necessary)
- Add feature of user groups to *cdd-tool*
- Start using common menu techniques in SkoleLinux



# Tasksel

- Debian-Edu supports tasksel in *education-tasks*
  - Conflicts with *cdd-common* from *experimental*
- Find a common solution



# Tasksel

- Debian-Edu supports tasksel in *education-tasks*
- Conflicts with *cdd-common* from experimental

→ Find a common solution



# Tasksel

- Debian-Edu supports tasksel in *education-tasks*
  - Conflicts with *cdd-common* from *experimental*
- Find a common solution



# Configuration

- Common solution for pre-seeding
- Common solution for cfengine



# Configuration

- Common solution for pre-seeding
- Common solution for cfengine



- 1 Ready
  - History
  - CDD Tool
  - Differences
- 2 **Steady**
  - CDD tools rewrite
  - **Web page generation**
- 3 Go





# Obtain web pages from tasks

- Tell people what *is* inside
- Web pages can be created from dependencies
- Package descriptions
- Make use of ddtp translated descriptions



# Obtain web pages from tasks

- Tell people what *is* inside
- Web pages can be created from dependencies
- Package descriptions
- Make use of ddtp translated descriptions



# Obtain web pages from tasks

- Tell people what *is* inside
- Web pages can be created from dependencies
- Package descriptions
- Make use of ddtp translated descriptions



# Obtain web pages from tasks

- Tell people what *is* inside
- Web pages can be created from dependencies
- Package descriptions
- Make use of ddtp translated descriptions



# Create database of future includes

- Tell people what *will be* inside
- Build *Packages.gz* like data base
- Translation is a problem
- Perhaps asking debian-www translation team
- Symbolic of traffic lights
  - Green – inside
  - Yellow – in future packages system
  - Red – should be packaged



# Create database of future includes

- Tell people what *will be* inside
- Build *Packages.gz* like data base
- Translation is a problem
- Perhaps asking debian-www translation team
- Symbolic of traffic lights

Red = Green = include

Yellow = Symbolic packages

Red = Should be package



# Create database of future includes

- Tell people what *will be* inside
- Build *Packages.gz* like data base
- Translation is a problem
  - Perhaps asking debian-www translation team
  - Symbolic of traffic lights



# Create database of future includes

- Tell people what *will be* inside
- Build *Packages.gz* like data base
- Translation is a problem
- Perhaps asking debian-www translation team
- Symbolic of traffic lights
  - Green = inside





# Create database of future includes

- Tell people what *will be* inside
- Build *Packages.gz* like data base
- Translation is a problem
- Perhaps asking debian-www translation team
- Symbolic of traffic lights
  - Green = inside
  - Yellow = inofficial packages outside
  - Red = Should be packaged



# Create database of future includes

- Tell people what *will be* inside
- Build *Packages.gz* like data base
- Translation is a problem
- Perhaps asking debian-www translation team
- Symbolic of traffic lights
  - Green = inside
  - Yellow = inofficial packages outside
  - Red = Should be packaged



# Create database of future includes

- Tell people what *will be* inside
- Build *Packages.gz* like data base
- Translation is a problem
- Perhaps asking debian-www translation team
- Symbolic of traffic lights
  - Green = inside
  - Yellow = inofficial packages outside
  - Red = Should be packaged



# Create database of future includes

- Tell people what *will be* inside
- Build *Packages.gz* like data base
- Translation is a problem
- Perhaps asking debian-www translation team
- Symbolic of traffic lights
  - Green = inside
  - Yellow = inofficial packages outside
  - Red = Should be packaged



# Create database of future includes

- Tell people what *will be* inside
- Build *Packages.gz* like data base
- Translation is a problem
- Perhaps asking debian-www translation team
- Symbolic of traffic lights
  - Green = inside
  - Yellow = unofficial packages outside
  - Red = Should be packaged
- Webtool to display bugs of dependant packages



# Future

- Base SkoleLinux on *cdd-tool* (to be rewritten)
- Enhance visibility on [www.debian.org](http://www.debian.org)
- Spread SkoleLinux to more countries
- Take over world domination



# Future

- Base SkoleLinux on *cdd-tool* (to be rewritten)
- Enhance visibility on [www.debian.org](http://www.debian.org)
- Spread SkoleLinux to more countries
- Take over world domination



# Future

- Base SkoleLinux on *cdd-tool* (to be rewritten)
- Enhance visibility on [www.debian.org](http://www.debian.org)
- Spread SkoleLinux to more countries
- Take over world domination





# Future

- Base SkoleLinux on *cdd-tool* (to be rewritten)
- Enhance visibility on [www.debian.org](http://www.debian.org)
- Spread SkoleLinux to more countries
- Take over world domination

