
A smart way of customizing Free Software

The example of Custom Debian distributions.

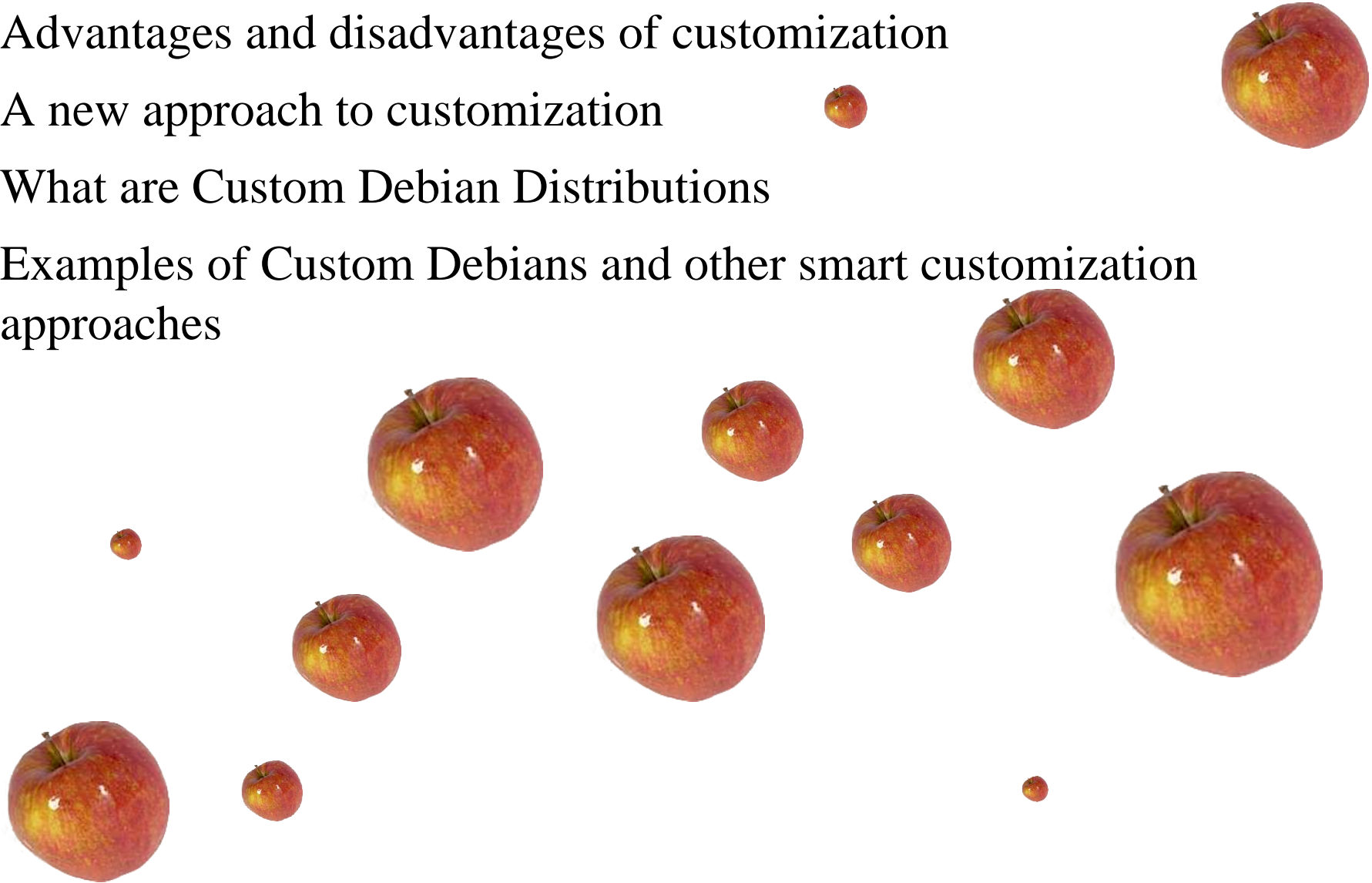
May 5, 2005

27 slides

Enrico Zini (enrico@debian.org)

What I'll be talking about

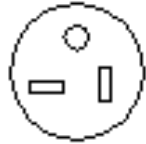
- 1) Advantages and disadvantages of customization
- 2) A new approach to customization
- 3) What are Custom Debian Distributions
- 4) Examples of Custom Debians and other smart customization approaches



Customization



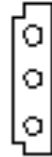
NEMA 5-15P
15A 125V



NEMA 5-20P
20A 125V



NEMA 6-15P
15A 250V



Mate-N-Lok



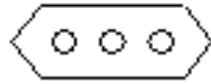
AS3112



BS 1363



CEE 7/VII



CEI 23-16/VII



DEMKO
107/10



JIS 8303



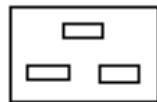
SEV 1011



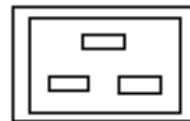
IEC 60320-1
C13 Plug
C14 Inlet



Reverse
IEC 60320-2-2
Sheet E Plug
Sheet F Inlet



IEC 60320-1
C19 Plug
C20 Inlet



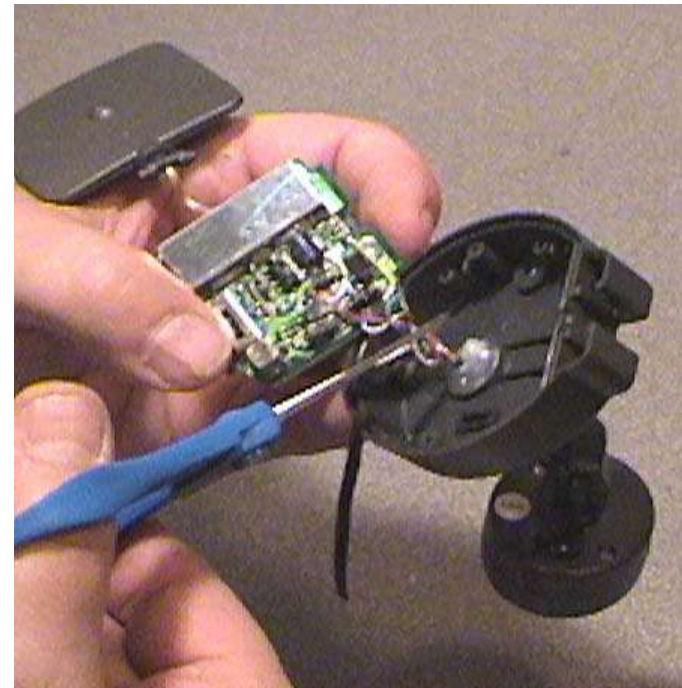
Reverse
IEC 60320-2-2
Sheet I Plug
Sheet J Inlet

Everyone has different needs.

Advantages of customization

Software is more appropriate and more efficient

- It has less unneeded functionality
- It is easier to document and support
- It requires less time to be adapted
- It can have a specialised user community
- It can talk a specific language or jargon
- It does what **you want right after you install it**
- **Even the installer does what you want :)**



Problems of customization

Customized software has a higher cost

- Know-how is harder to find
- Support is harder to find
- The development community is smaller
- There may not be enough resources to develop it further
- One-time costs are shared by a smaller number of entities

The main cause of these problems is divergency from mainstream development



Advantages without disadvantages

With Free Software there is a way beyond the dilemma:
customization without diverging!

- Existing software can be selected to build a custom system
- Existing software can be configured to be a part of a custom system
- Existing software can be extended to include needed features
- Existing software can be made configurable to exclude unneeded features
- All of these things can be done as a part of the main developer community



The Custom Debian approach to creating operating systems

also known as:

The last, final step towards Total World Domination!

Debian GNU/Linux

16302 "packages" (programs, code libraries, documentation...)

920 volunteer developers from 50 different nations

11 supported architectures

Open self-organised community with constitution, elections, votes...

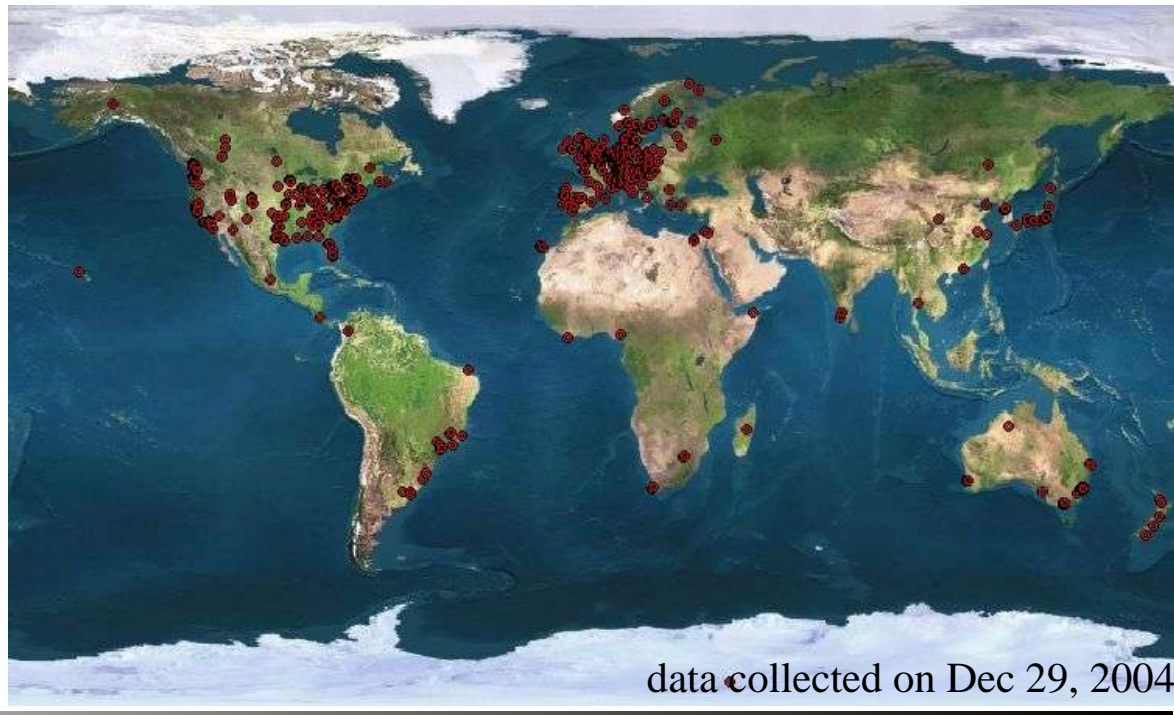
Infrastructure dedicated to community services and groupwork

10 years old



Problem: it's BIG.

So big that we begun developing "custom" versions as subsets for specific needs.



Definition of CDD

Custom Debian Distributions:
distributions derived from Debian
which are still 100% Debian

(successful revolutions need simple ideas)

CDD HOWTO

- 1) Take Debian
- 2) Select Packages
- 3) Configure Packages
- 4) Rule the world

Aim at being 100% policy compliant

If you need special software, package it in Debian

If you need special configurations, work with maintainers

If you need stable software, fix bugs and submit patches to the BTS

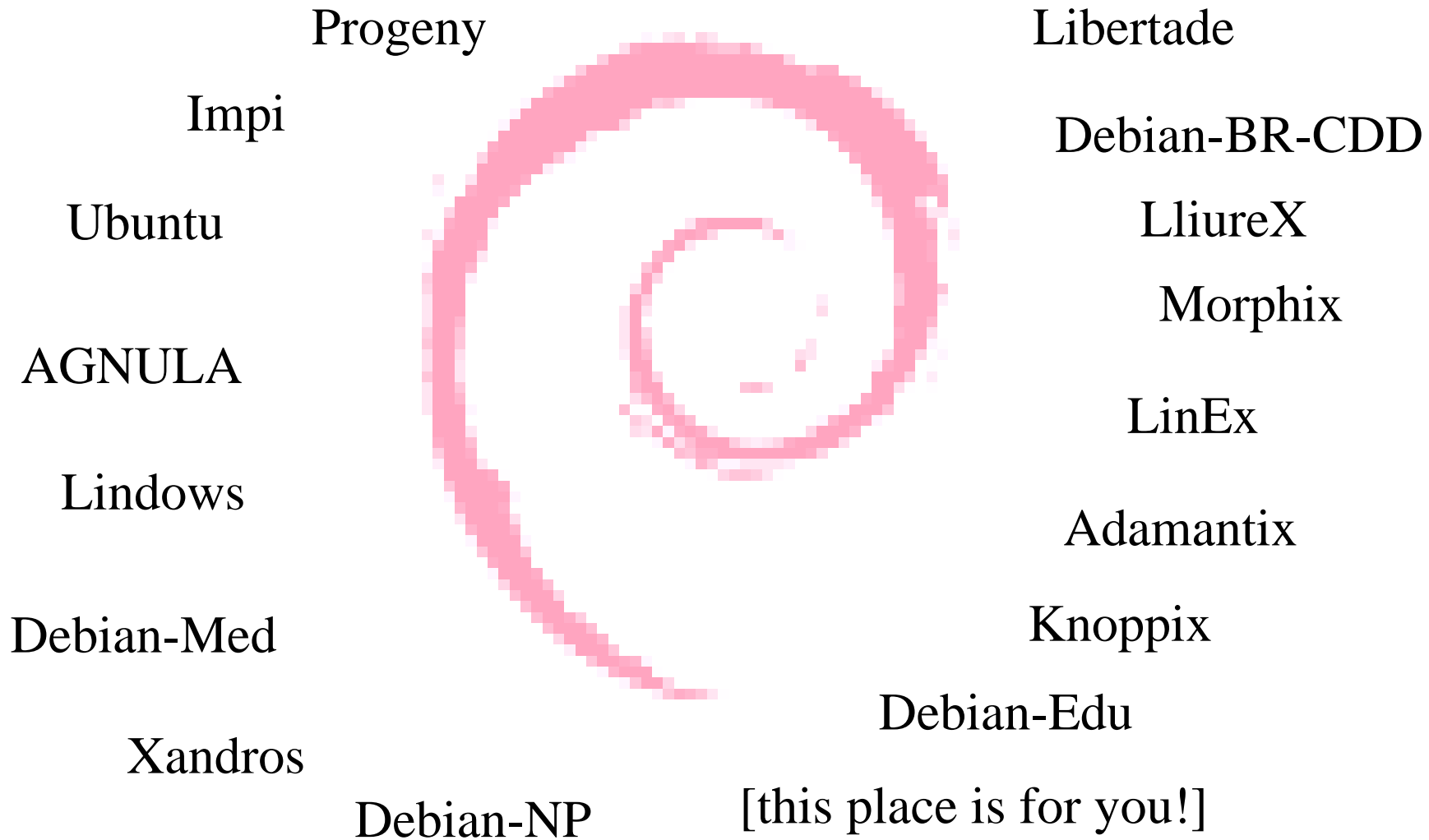
If you need translations, work with Debian translators

If you need security, work with the Debian security team

(when it works like this, the World Takeover is complete)

Why it works

Various Debian-derived systems...



...can coexist inside Debian!



Progeny [this place is for you!]
Impi Morphix
Ubuntu Debian-BR-CDD
AGNULA LliureX
Lindows Knoppix Libertade
Debian-Med Adamantix
Xandros LinEx
Debian-NP
Debian-Edu

First magic of CDDs

**Once you cooperate with Debian
You cooperate with all the others**

(successful revolutions are made by lazy revolutionaries)

Extreme Customization

Debian is a wad of wet clay: CDDs give it a shape!

With CDDs you finally can:

- Work with what you like!
- Choose your users!
- Market to specific groups
- Provide (sell?) user support
- Foster specific user communities
- Create targeted documentation!
- Solve specific problems!



Make people (and yourself) satisfied and happy!

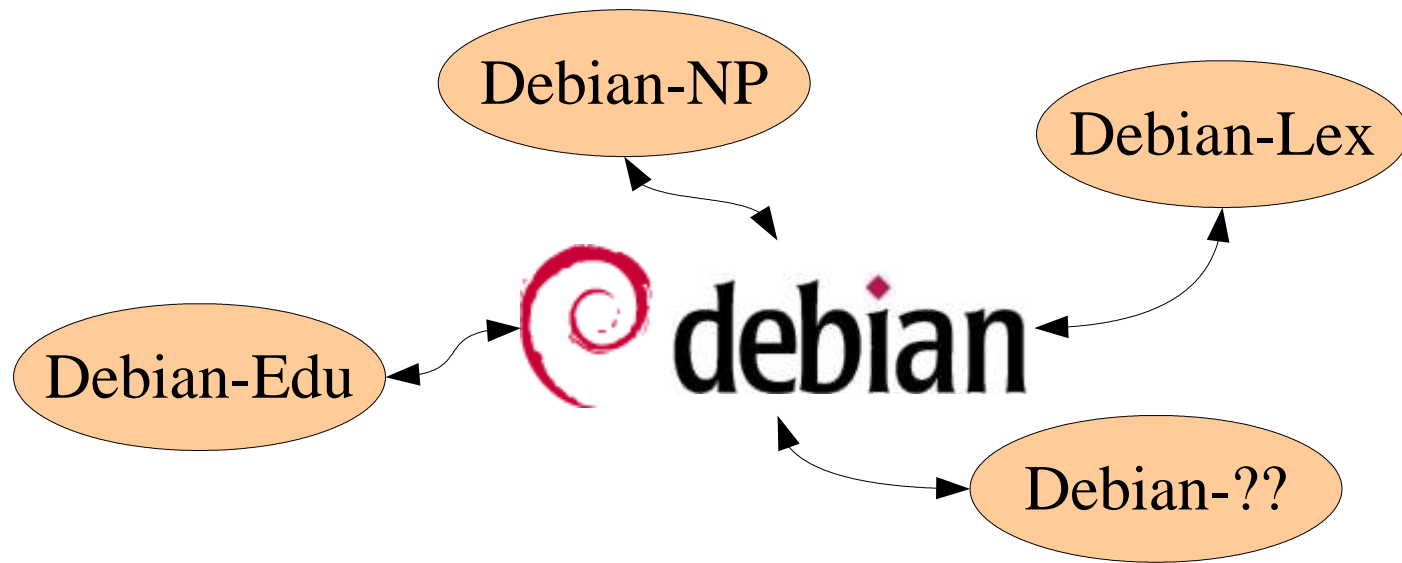
(successful revolutions happen when people are happy with it)

Custom Debian

Who makes Custom Debians contributes to Debian, and Debian improvements contribute to all Custom Debians.

When you do Free Software, you create *externalities*. With CDDs they are collected inside Debian, where they "fermentate" using powerfully creative *network economy* processes.

Everyone drinks the wine!



Summary so far

It is possible to do customization while minimizing divergency.

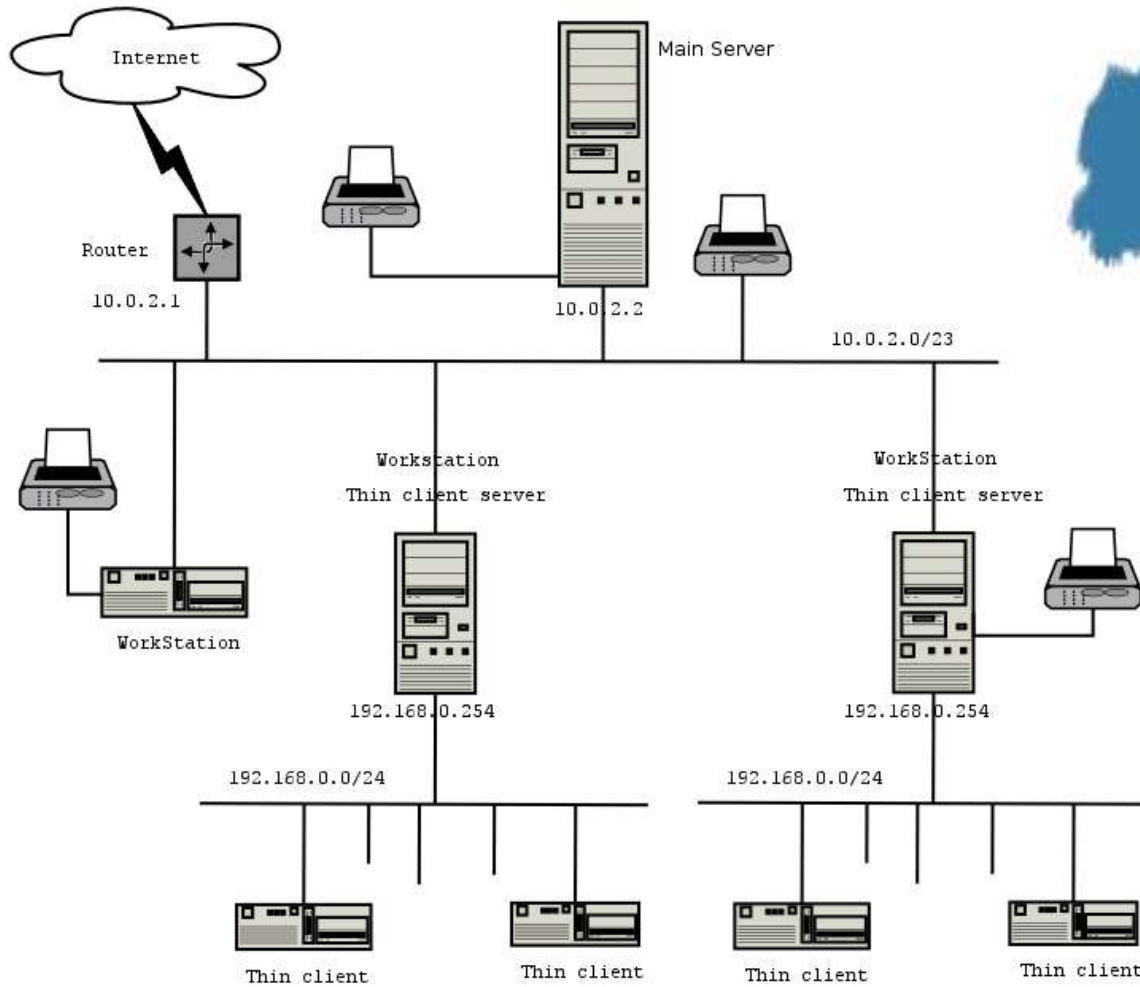
- It allows to customize without losing quality and progress
- It brings you external feedback and know-how
- It increases the possibilities of innovation
- It can put you in touch with groups with the same needs for customization

This has many advantages, but it may require some changes on how we work



Some examples

Debian-Edu/Skolelinux



Started in Norway in 2001, then merged with Debian-Edu.

3 years later, in various reports: "the only computer solution that takes the schools' needs and resources seriously" (Statskonsult report #18, Dec 2003)

Debian-Edu/Skolelinux

From Joey Hess' TODO-list for SkoleLinux:

We want Sarge to release as soon as possible, including all the packages Debian Edu want and need to be able to release the next major release of Debian Edu with packages only from Sarge.

To be able to release Debian Edu with package only from Sarge, we need to make sure:

- the packages in Sarge can be installed out of the box with the configuration we want to use in Debian Edu
- all the packages we want are included in Sarge



Other Custom Debians



debian♦np

Debian-Med

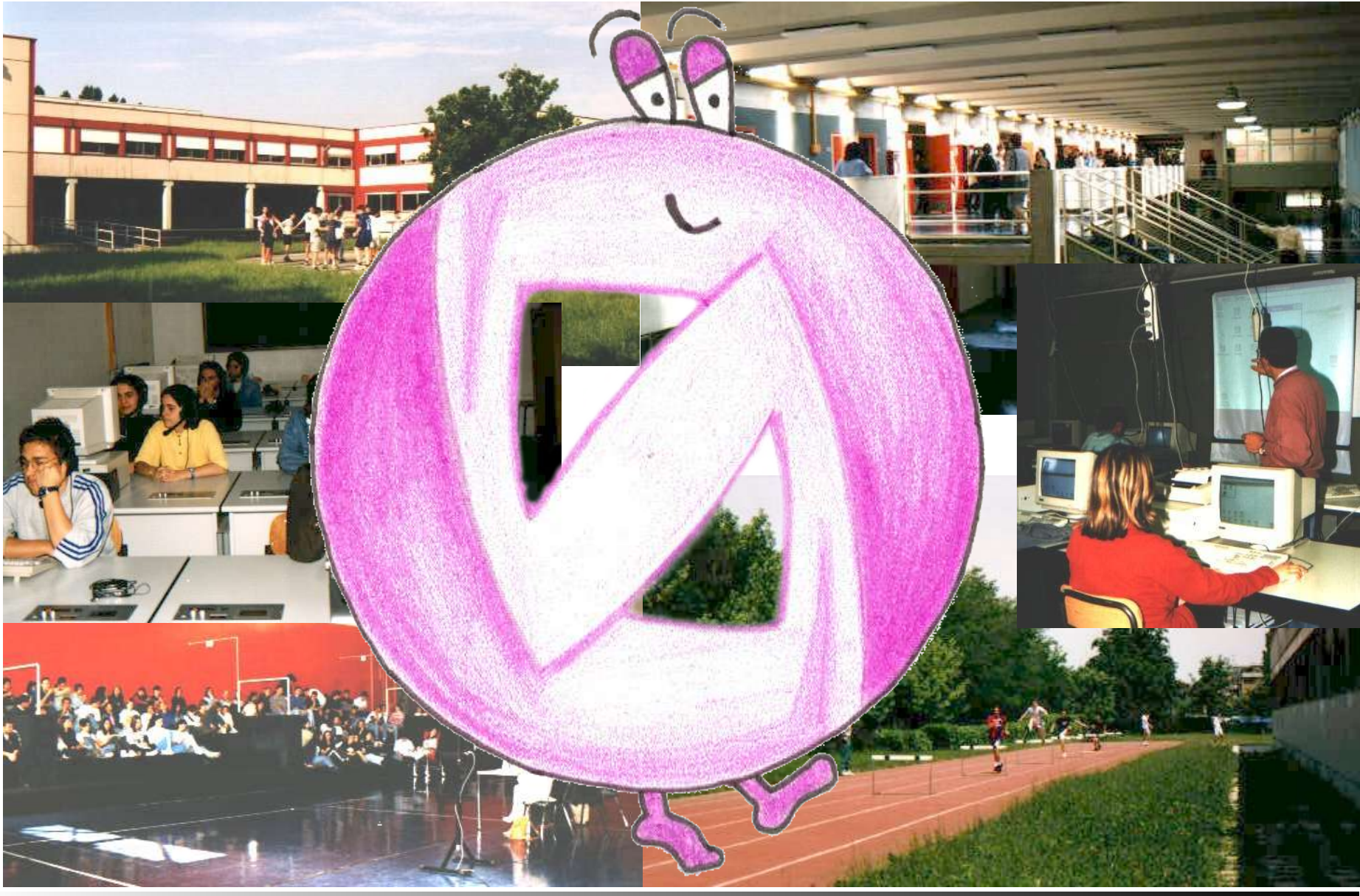
**Some other examples
which are not CDDs,
but they get the idea**

Ubuntu



Ubuntu is diverging from Debian, but tries to converge again every 6 months (after every release).

The Italian OpenOffice.org Thesaurus



Some suggestions

Have in the team someone who is an active part of the community.

- They already know how things work, technically and socially
- They can mediate between the community and the other developers
- They can keep local modification updated and in line with new development
- They can bring in new know-how
- They can keep the project networked with other similar groups, and help forming synergies

Distribute your software as part of Debian.

- It can be brought to a wider audience making it more useful
- It can be brought to a wider audience making it more tested

Conclusion

Creating and maintaining your custom high-quality operating system has never been that easy (and it will become easier and easier).

You can help by improving the existing CDD infrastructure.

You can help by merging your own Debian fork back into Debian as a CDD.

You can help by starting your own CDD (and I have ideas for you).

Resources:

- `debian-custom@lists.debian.org`
- `#debian-custom` on Freenode

Question time.