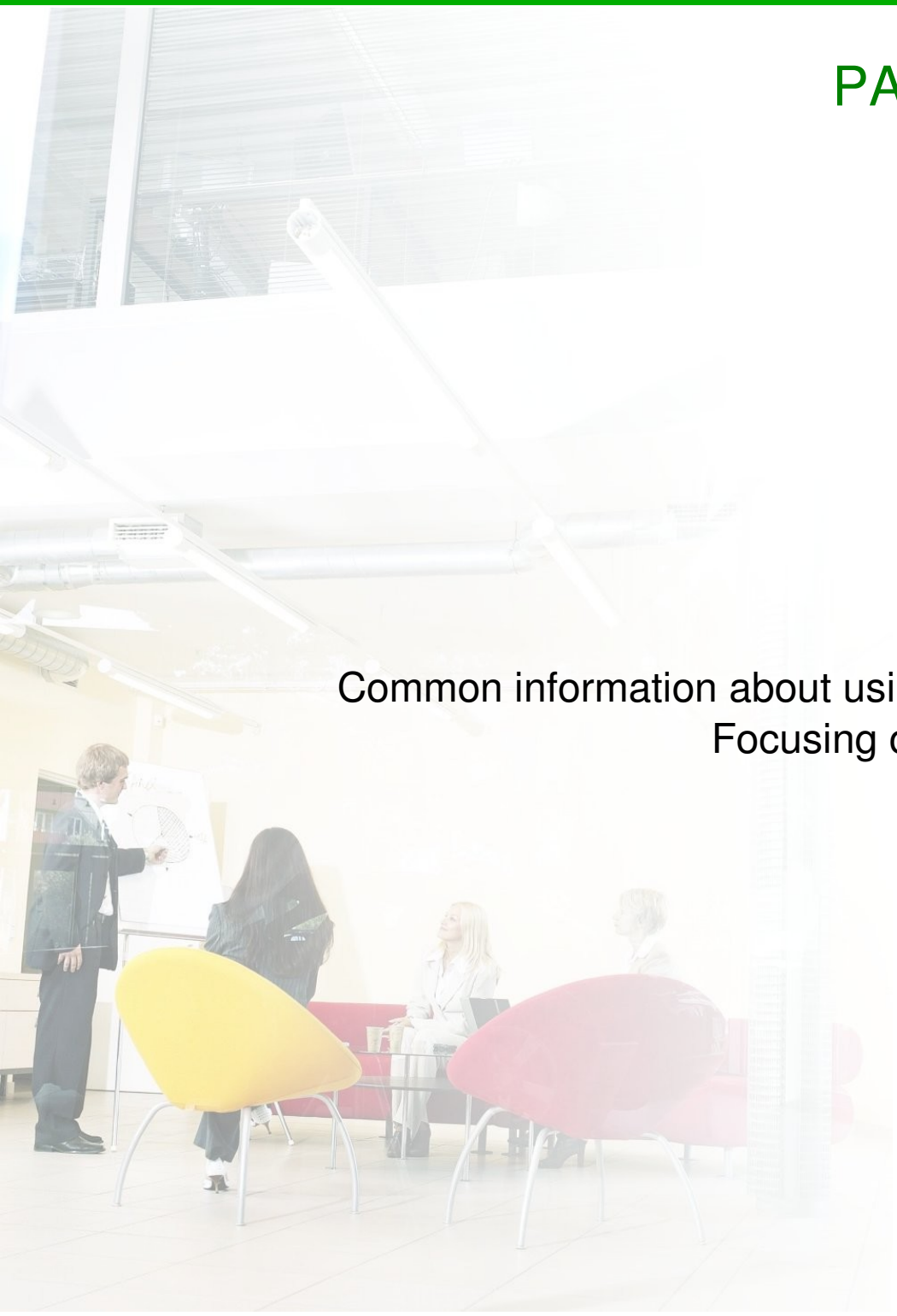
Embedded Linux

Miko Nieminen

Nomovok Ltd.
+358408228047
miko.nieminen@nomovok.com
http://www.nomovok.com

NOMOVOK

PART 1



Common information about using Linux in embedded systems.
Focusing on mobiles.

NOMOVOK

Basics Concepts

- Distributions: Redhat, SuSE, Debian, Ubuntu (variant of Debian), Gentoo, ...
- Linux: basically just kernel.
- Mobiles: subset of embedded systems (in this representation), having graphical user interface and moderate resources.
- OSS: Open Source Software (also FOSS, FLOSS)
- Desktop environments: Gnome, KDE, Xfce
- SDK: Software Development Kit

NOMOVOK

Using Linux in Embedded Systems

Examples

- Internet tablets
- mobile phones
- navigation systems
- set-top boxes
- networking devices

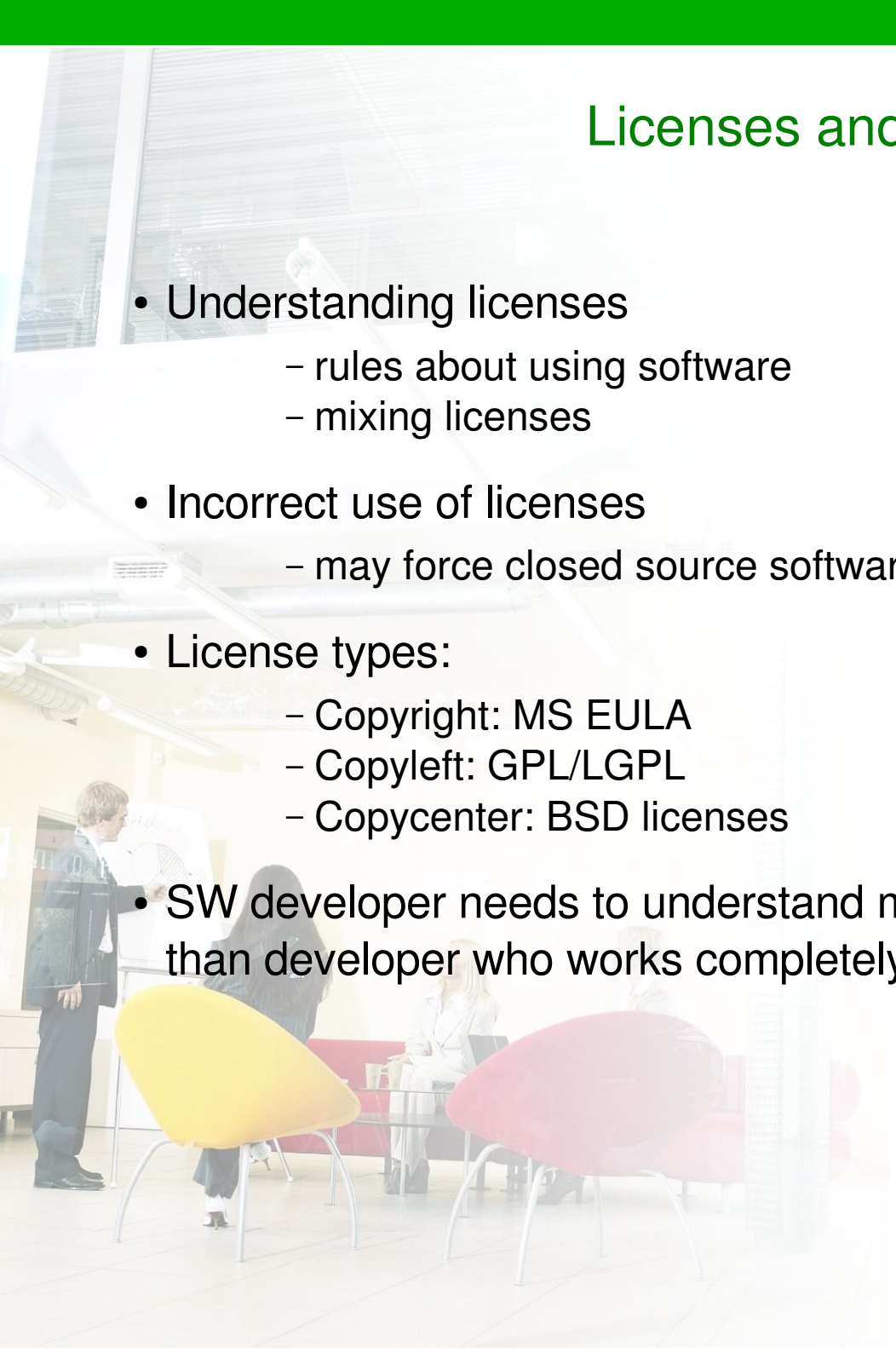
Problematic areas

- Availability of low level drivers
  - radios
  - accelerators for graphics
  - DSPs
  - power management
- Closed source UI design
  - usability patents
  - own look&feel
- Bringing OSS into closed source systems
  - Python in Series60

NOMOVOK

Licenses and Legal Aspects

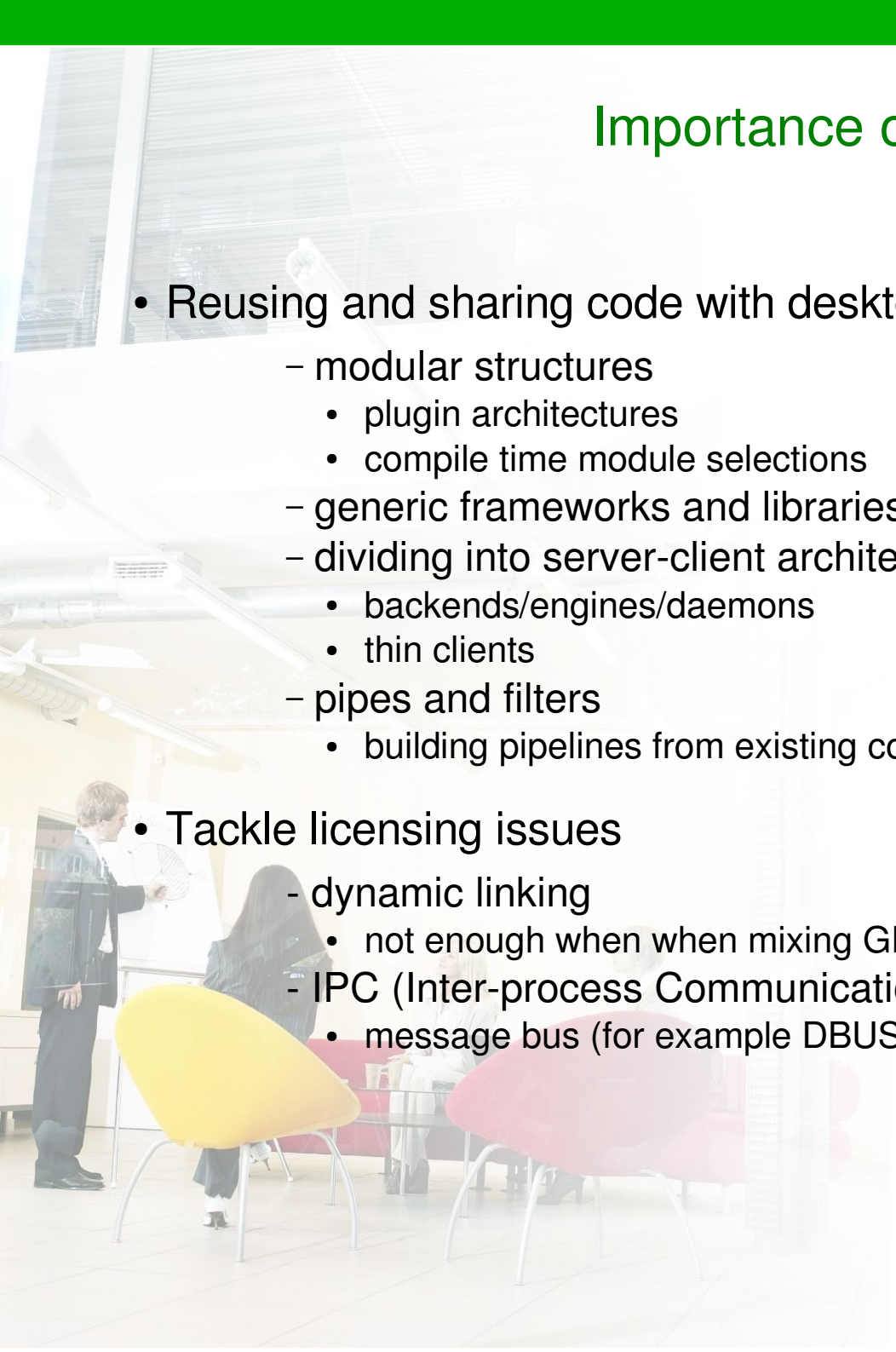
- Understanding licenses
  - rules about using software
  - mixing licenses
- Incorrect use of licenses
  - may force closed source software to be opened
- License types:
  - Copyright: MS EULA
  - Copyleft: GPL/LGPL
  - Copycenter: BSD licenses
- SW developer needs to understand more about licensing and legal issues than developer who works completely in proprietary world!



NOMOVOK

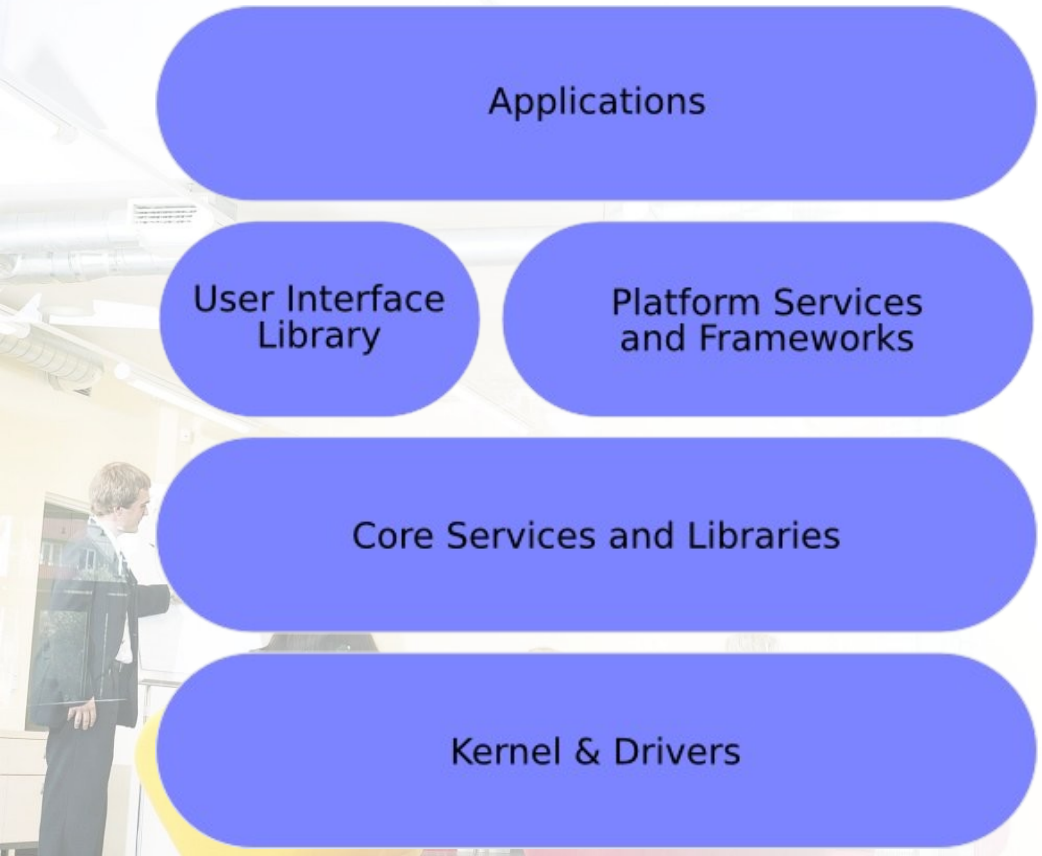
Importance of Architectures

- Reusing and sharing code with desktop systems
  - modular structures
    - plugin architectures
    - compile time module selections
  - generic frameworks and libraries
  - dividing into server-client architectures
    - backends/engines/daemons
    - thin clients
  - pipes and filters
    - building pipelines from existing components without writing new ones
- Tackle licensing issues
  - dynamic linking
    - not enough when when mixing GPL and proprietary software
  - IPC (Inter-process Communication)
    - message bus (for example DBUS)



NOMOVOK

High Level Application Stack



Kernel & Drivers

- Low level components, hardware specific layer

Core Services and Libraries

- Low level libraries (including low level graphical components)
- Mostly components defined by LSB (Linux Standard Base)

User Interface Library

- Toolkits and helper libraries for making UI design easier

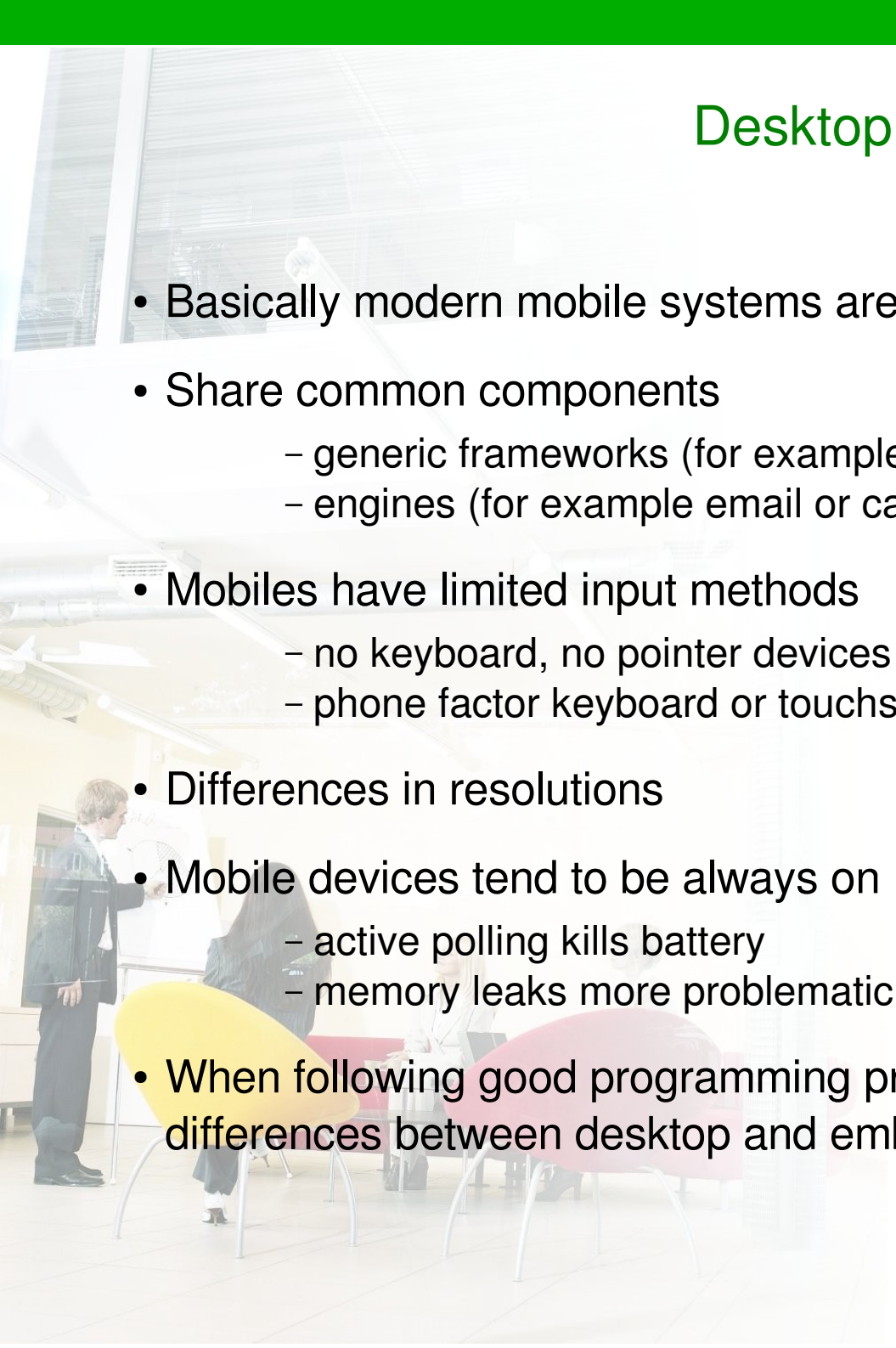
Platform Services and Frameworks

- Product or product family specific middleware

NOMOVOK

Desktop vs. Mobile

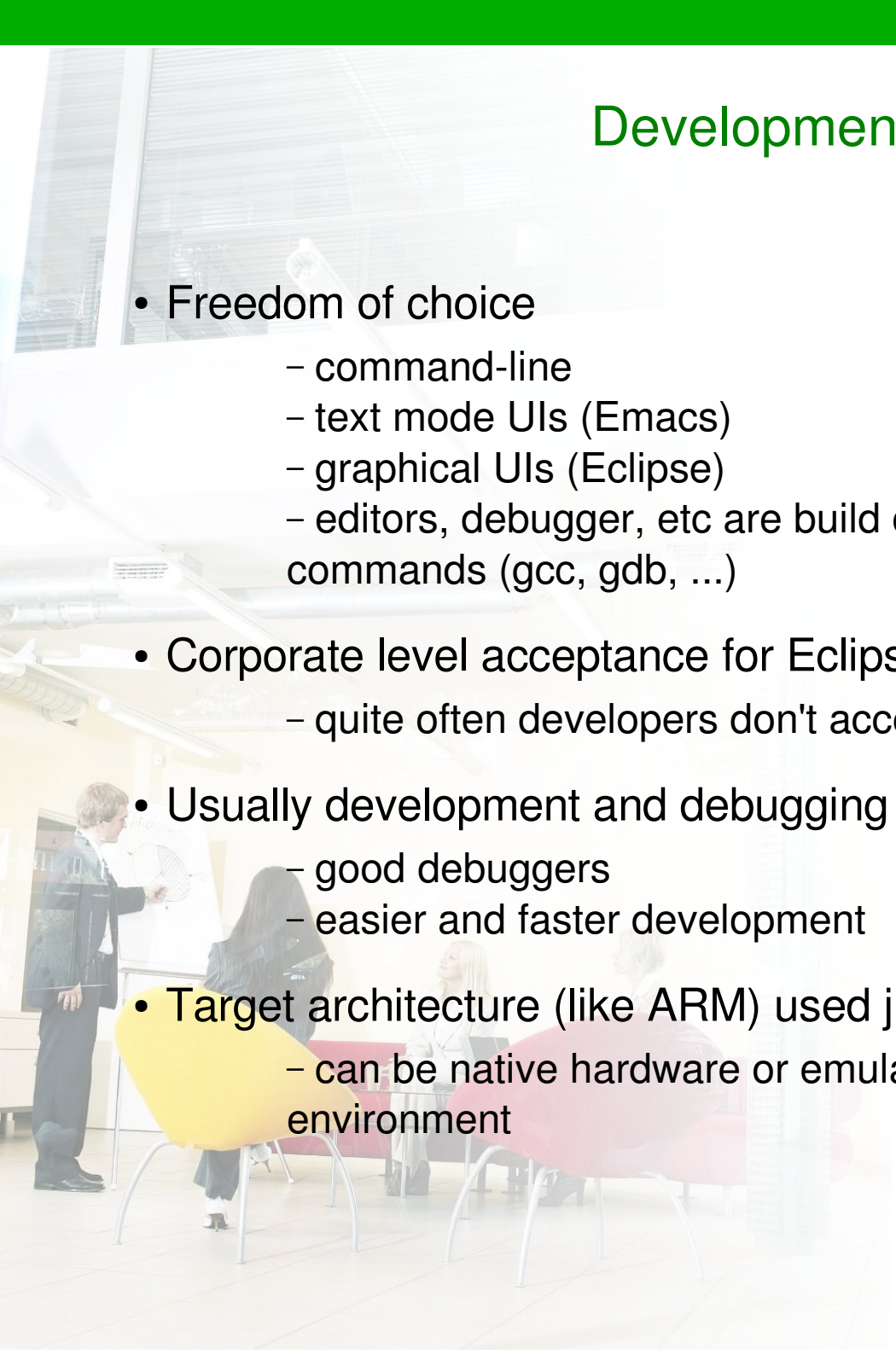
- Basically modern mobile systems are very near full desktop environments.
- Share common components
  - generic frameworks (for example multimedia)
  - engines (for example email or calendar)
- Mobiles have limited input methods
  - no keyboard, no pointer devices (no counting touchscreen here)
  - phone factor keyboard or touchscreen
- Differences in resolutions
- Mobile devices tend to be always on
  - active polling kills battery
  - memory leaks more problematic
- When following good programming practices, there aren't that many differences between desktop and embedded software development.



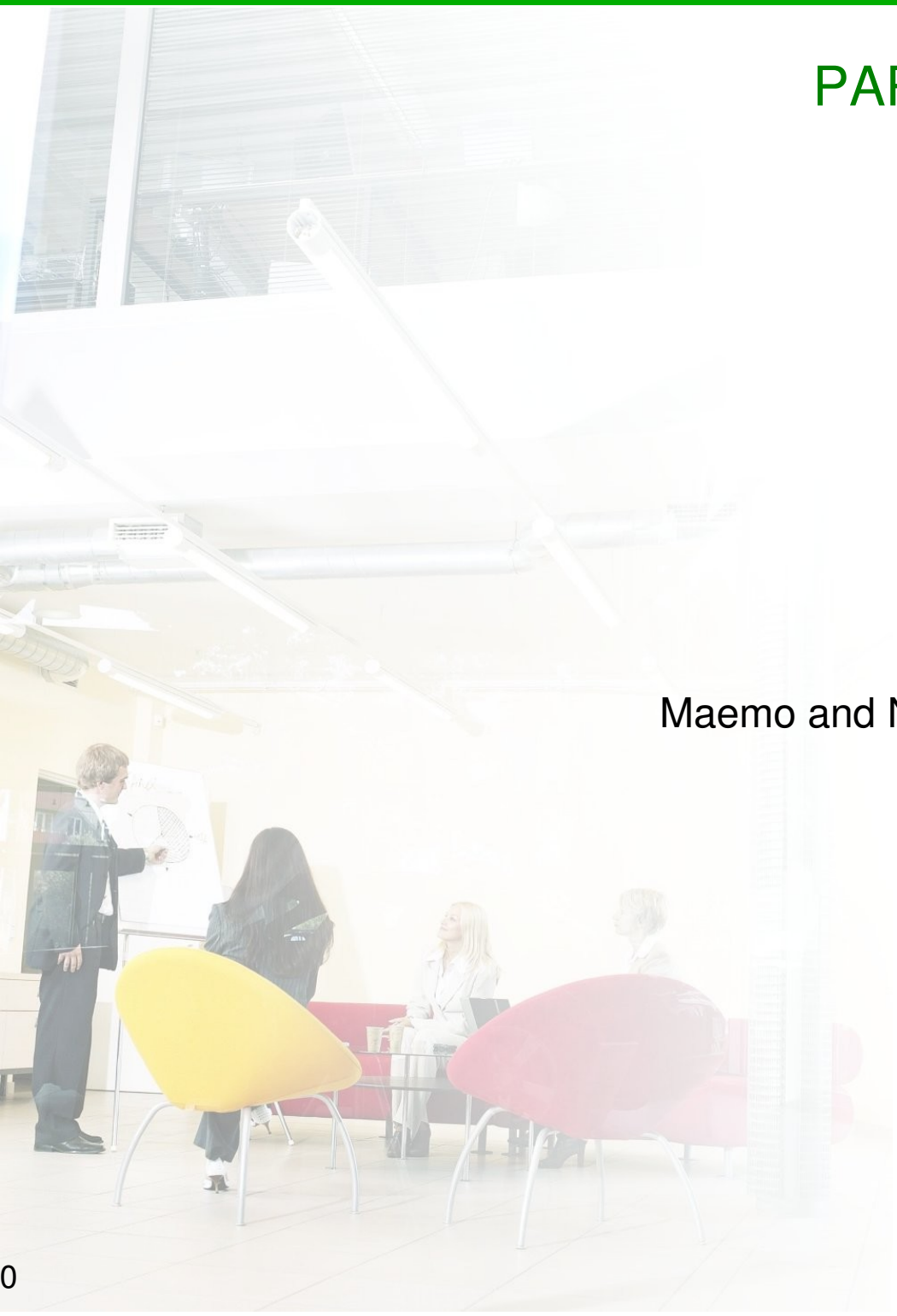
NOMOVOK

Development Environments

- Freedom of choice
  - command-line
  - text mode UIs (Emacs)
  - graphical UIs (Eclipse)
  - editors, debugger, etc are build on top of “standard” libraries and commands (gcc, gdb, ...)
- Corporate level acceptance for Eclipse
  - quite often developers don't accept
- Usually development and debugging done in x86 based environment
  - good debuggers
  - easier and faster development
- Target architecture (like ARM) used just for building software
  - can be native hardware or emulated environment or cross compilation environment



NOMOVOK



PART 2

Maemo and Nokia 770/800

NOMOVOK

What is Maemo

- Maemo is a community build around Nokia 770 and Nokia 800 products
- Under Maemo, Nokia provides SDK which is targeted for 770 and 800
- Maemo SDK and products are based on Debian Linux

NOMOVOK

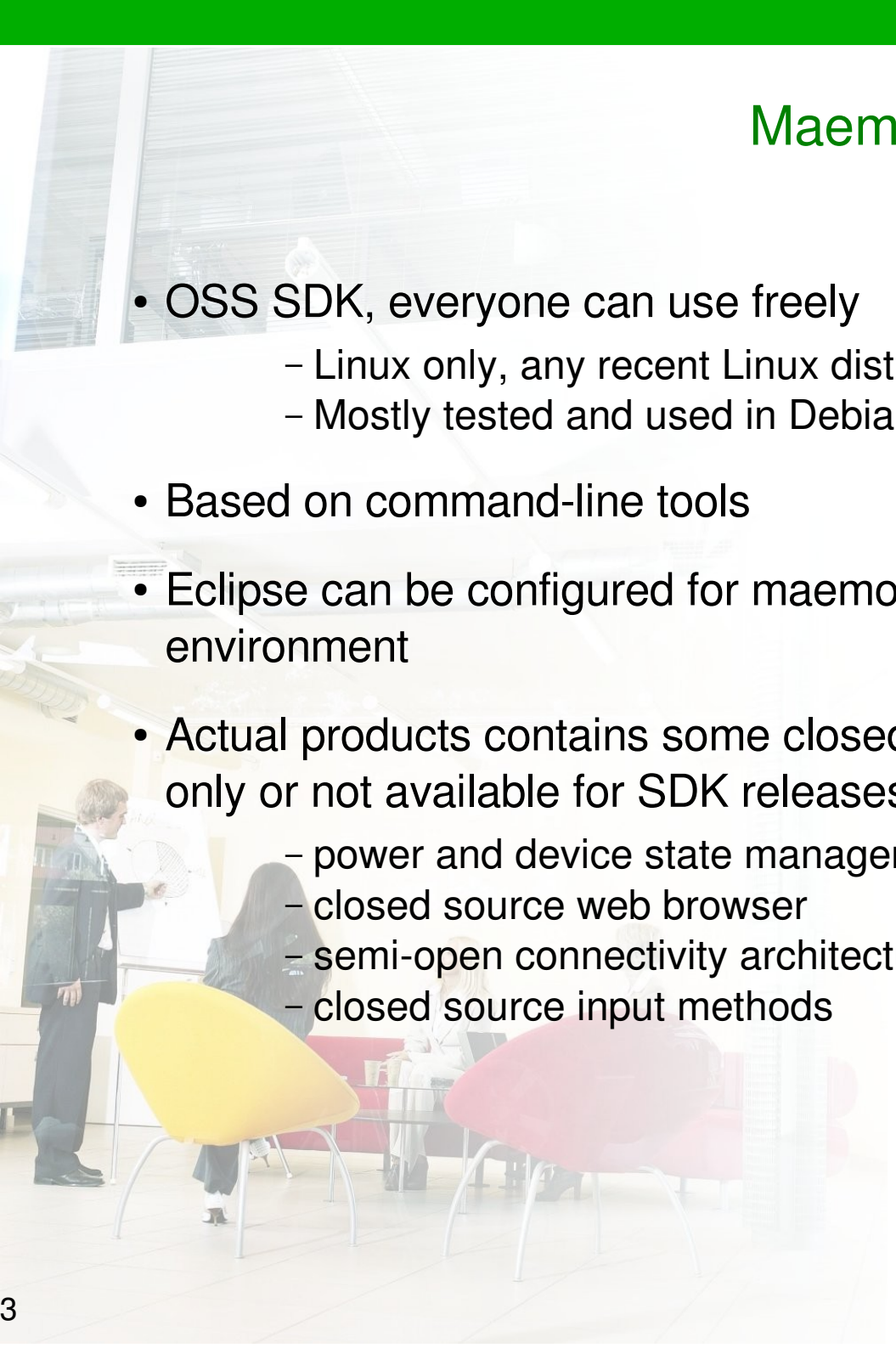
Maemo as Community

- Community for end users and developers
- For end users, Maemo provides better and faster support than most corporate support services and support comes directly from developers (Nokia or community)
- For developers, Maemo provides lots of documentation and SDK
- Of course Maemo provides means for communication, in this case mailing lists and wiki. Also web-based forum can be found from Internet Tablet Talk (I think it is 3rd-party site, but not sure)

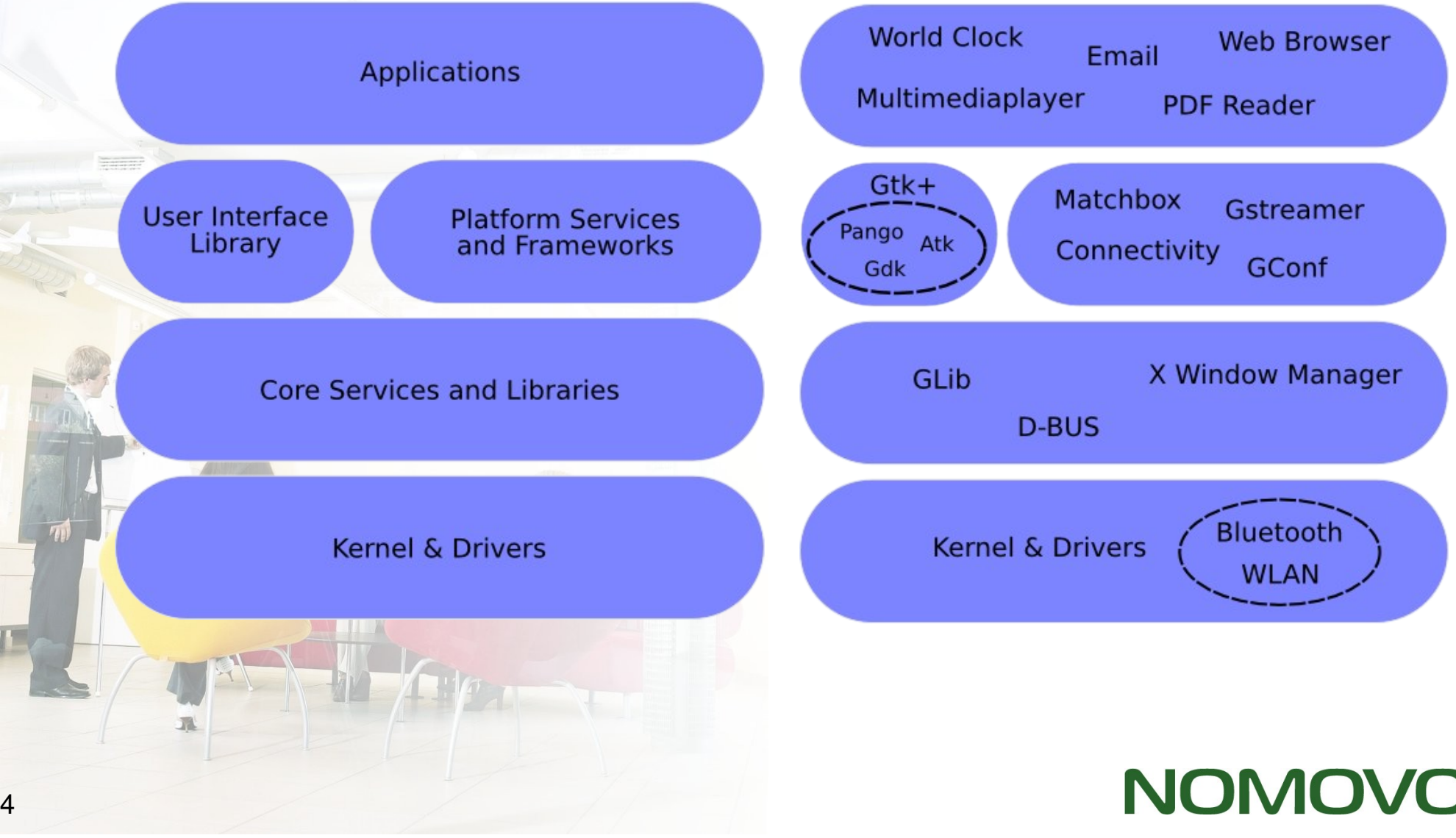
NOMOVOK

Maemo SDK

- OSS SDK, everyone can use freely
  - Linux only, any recent Linux distributions should work
  - Mostly tested and used in Debian, Ubuntu and Redhat
- Based on command-line tools
- Eclipse can be configured for maemo sdk to have graphical development environment
- Actual products contains some closed source components which are binary only or not available for SDK releases:
  - power and device state management framework
  - closed source web browser
  - semi-open connectivity architecture
  - closed source input methods



High Level Architecture



Generic

Applications

User Interface Library

Platform Services and Frameworks

Core Services and Libraries

Kernel & Drivers

Maemo

World Clock Email Web Browser

Multimediaplayer PDF Reader

Gtk+

Pango Atk Gdk

Matchbox Gstreamer

Connectivity GConf

GLib X Window Manager

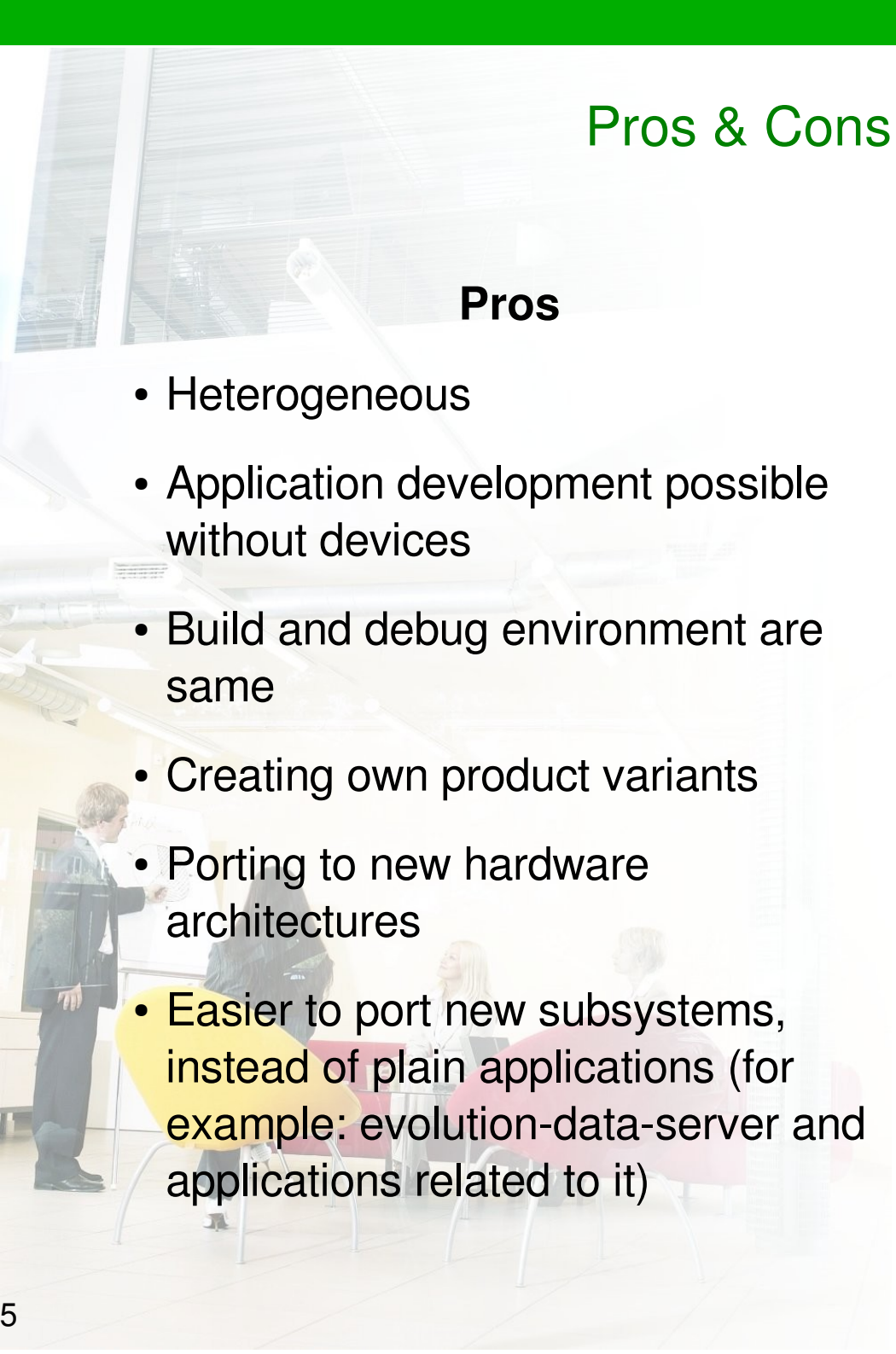
D-BUS

Kernel & Drivers

Bluetooth WLAN

NOMOVOK

Pros & Cons of Open SDK



Pros

- Heterogeneous
- Application development possible without devices
- Build and debug environment are same
- Creating own product variants
- Porting to new hardware architectures
- Easier to port new subsystems, instead of plain applications (for example: evolution-data-server and applications related to it)

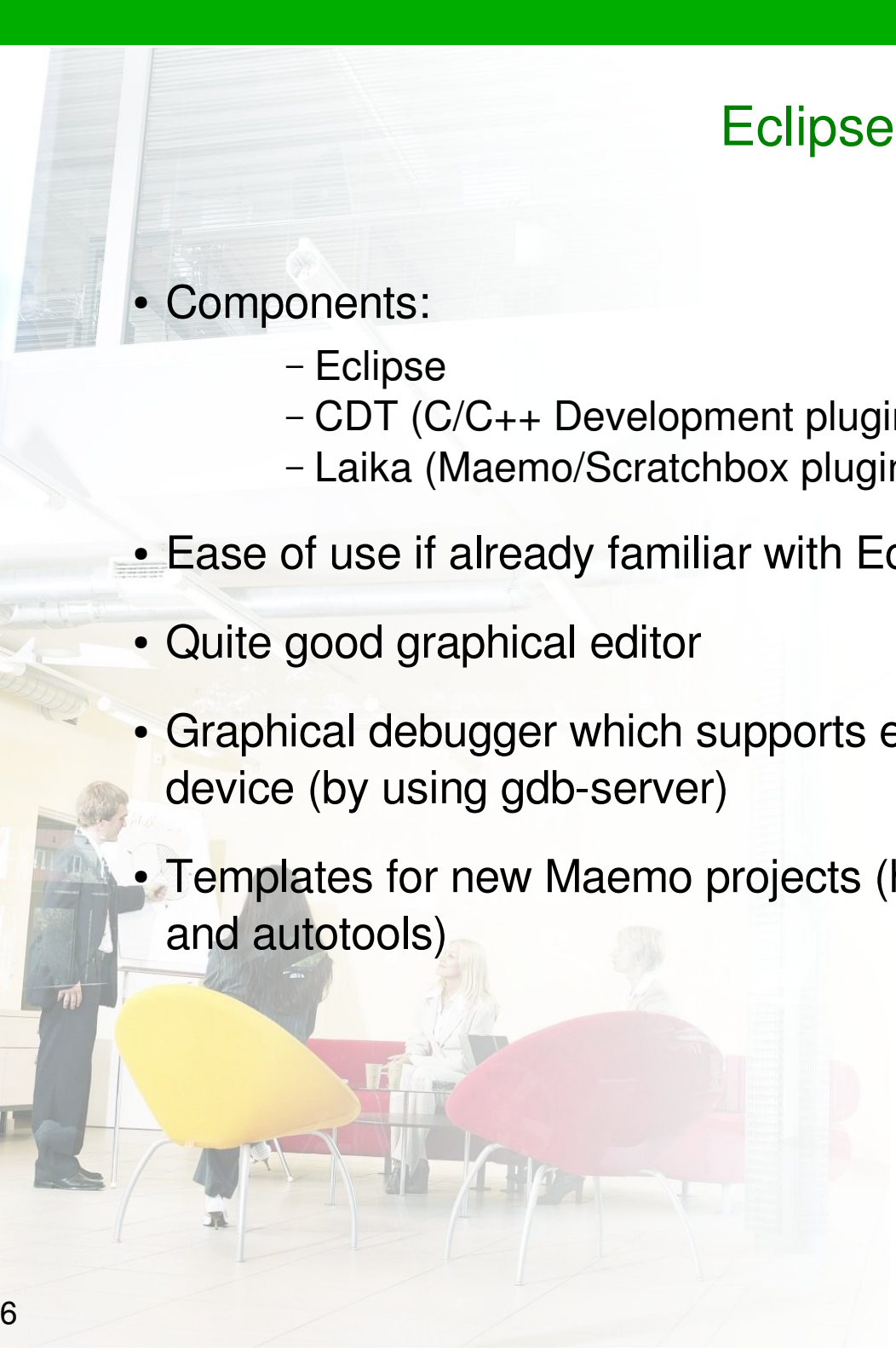
Cons

- Heterogeneous
- Productizing SDK is more complicated, because it is build from multiple OSS projects
- Lots of moving parts -> may cause inconsistencies between different users' environments

NOMOVOK

Eclipse & Laika

- Components:
  - Eclipse
  - CDT (C/C++ Development plugin for Eclipse)
  - Laika (Maemo/Scratchbox plugin for Eclipse)
- Ease of use if already familiar with Eclipse
- Quite good graphical editor
- Graphical debugger which supports even remote debugging on the actual device (by using gdb-server)
- Templates for new Maemo projects (hiding some hard parts of packaging and autotools)



NOMOVOK

PART 3

Conclusions, future of embedded linux and further reading.

NOMOVOK

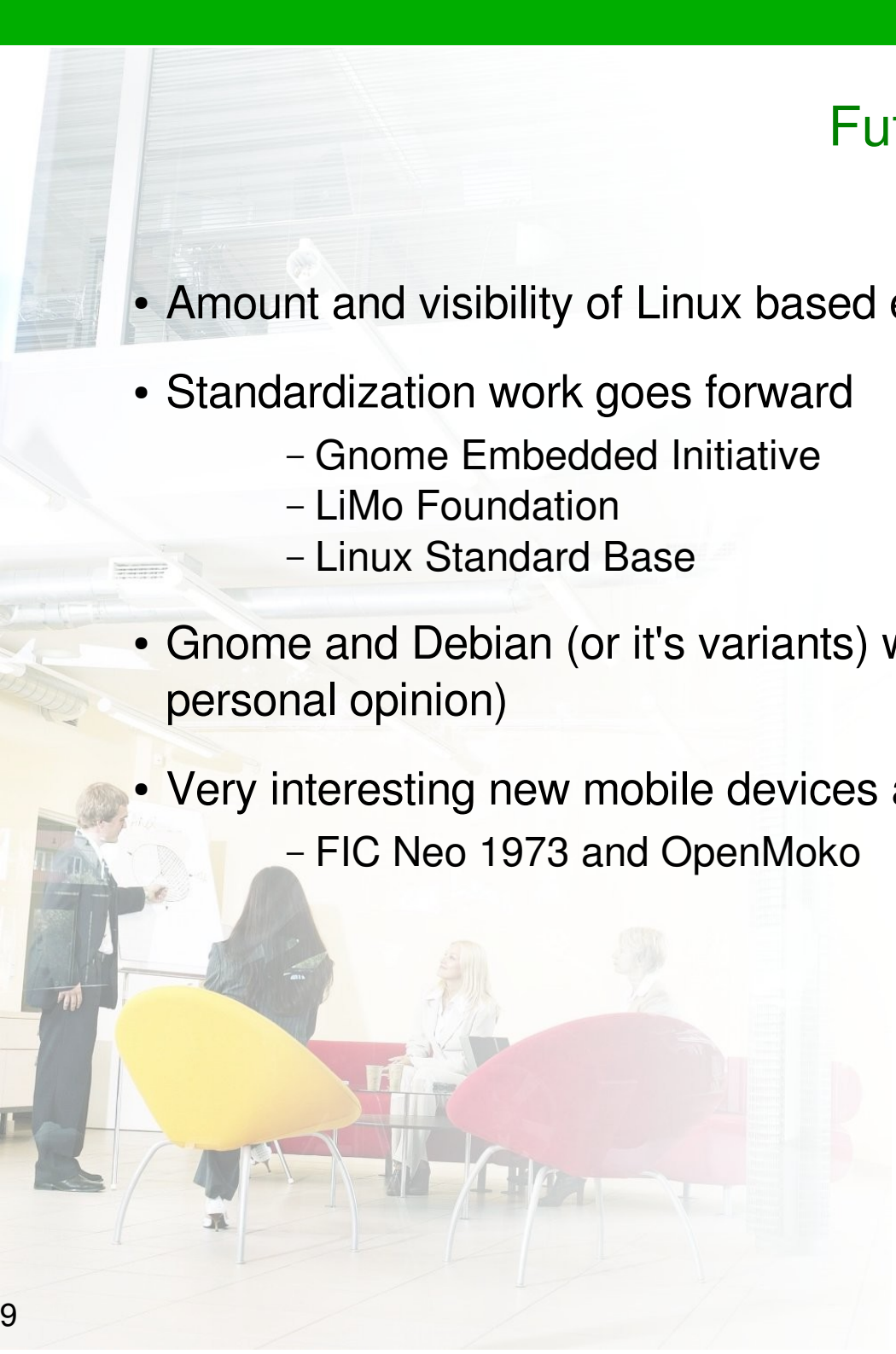
Conclusions

- Linux can be places where you don't realize it -> more Linux devices than one could expect
- Platforms aren't standardized yet, but work toward common platforms is going forward
- SDK world is a bit messy (many ways to get same results), but doesn't bind developers as much as proprietary development environments
- Communities are very active

NOMOVOK

Future

- Amount and visibility of Linux based embedded solutions will increase.
- Standardization work goes forward
  - Gnome Embedded Initiative
  - LiMo Foundation
  - Linux Standard Base
- Gnome and Debian (or it's variants) will be most probable winners (my personal opinion)
- Very interesting new mobile devices and concepts
  - FIC Neo 1973 and OpenMoko



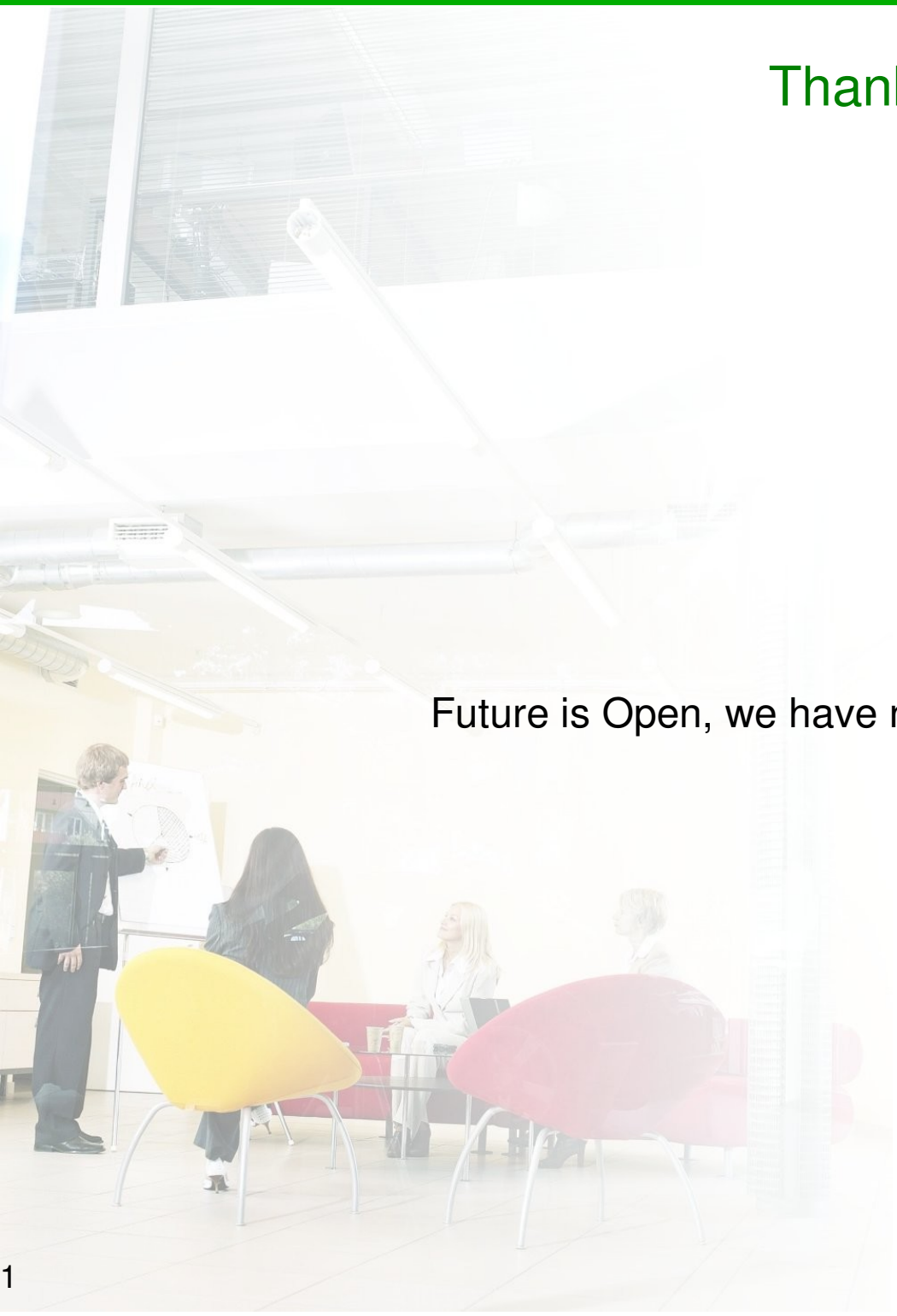
NOMOVOK

Links

- Linux Foundation and Linux Standard Base:
  http://www.linux-foundation.org/
  http://www.linux-foundation.org/en/LSB
- Maemo:
  http://maemo.org/
- Handhelds.org (one of the first embedded linux communities):
  http://www.handhelds.org/geeklog/index.php
- mdeb.org (information backbone for building Debian based embedded products, currently very new):
  http://www.mdeb.org/
- Internet Tablet Talk:
  http://www.internettablettalk.com/

NOMOVOK

Thank You!



Future is Open, we have no idea what we are doing!

NOMOVOK