Marketing Gtk+
Marketing: Understanding Choice
Core assets

Consistent and Stable API
Core assets

Large Binding Set
Core assets

Cross platform!
Core assets

New brand!
(not just logo)
We're quite sexy, don't we?
Some outsiders' quotes

"We eventually went for Qt for its documentation quality, consistent API and Windows support.

Jean-Baptiste Kempf, VLC"

"If I had to make the decision today, I would use Gtk+ for an application that is Linux-focused, not for anything cross-platform."

Justin Karneges, PSI
Common arguments against Gtk+

- C Sucks
- Bindings are just independent projects
-Gtk is ugly
- All themes look the same
- Ugly on Win32
- No native Mac port
But if we agree we are sexy...
... what's wrong with us then?
**Documentation**

*The Tutorial*

- Most obvious choice for beginners
- Not suitable to learn to write apps
- Code snippets not readable
- Rather Gtk Widget Manual
- Bad inspiration for bindings docs (gtkmm docs for the win)
- Assumes a bash like environment

*GTK+ 2.0 Tutorial*
Documentation

Reference manuals: library.gnome.org

- Great tool btw! (kudos to fpeters)
- Plain list of ALL modules
- Order doesn't match relevance
- See mobile.gnome.org
Bindings

What is GTK+?

GTK+ is a highly usable, feature rich toolkit for creating graphical user interfaces which boasts cross platform compatibility and an easy to use API. GTK+ is written in C, but has bindings to many other popular programming languages such as C++, Python and C# among others. GTK+ is licensed under the GNU LGPL 2.1 allowing development of both free and proprietary software with GTK+ without any license fees or royalties.

Get an overview of GTK+. Understand who started it, the basic architecture and why we use the license we do.

GTK+ has been involved in many projects and some big platforms. To see
Bindings

Unify the brand and websites of certified bindings
Distribution

Getting Gtk+ into the hand of developers, sysadmins and users

- Linux, OpenSolaris and BSD are doing fine
- No official installers on Windows or Mac (yet)
- Bad defaults
- Third party apps and binding are hard
The PyGTK All-in-One story
Distribution

Gtk+ as a native SDK
Conclusions

- Really bad PR in the past
- Documentation needs love
- Cross platform support is essential
- No clear path to redistribute apps
- Gtk+ as a monolithic SDK
- Big audience out there waiting for it
Thank you!

Questions anyone?