The middleware problem:
a tale of abstraction

Dario Freddi

13th November, 2012
Hello
Code less. Create more. Deploy everywhere.
Code less
tick
Create more
damn, tick
Deploy everywhere?
Middleware
Why?
Platform
Complicated
Different developers
Different developers' Needs
play("""cat.wav"");
create(const ObjectPtr & source, Format format, quint64 amount, double rate, bool flush, bool intermediate, quint64 duration, bool eos);
Results?
A single API does not fit
Can't I just use the lower layer?
Platform
Give choice
Give choice
Do not force
Solutions?
Authentication is required to grant authorizations to other users

The application /usr/bin/systemsettings is attempting to perform an action that requires privileges. Authentication as one of the users below is required to perform this action.

- Sebastian Kuegler (sebas)

Password for sebas: 

- Remember authorization
- For this session only

Details >> OK Cancel
Framework
The problem
What about developers?
Support new framework
Support new framework
Lose compatibility
Support old framework
Support old framework

Rewrite code later, lose compatibility
Using Qt?
Support both
High-level API
KAuth::Action action("my.own.action");
action.execute("my.own");
Easy to use
Tailored around a specific use case
Insufficient for developing clients
A high level API shouldn't make sense for that
Bind twice
Client libraries

==

Qt bindings
Not needed on all platforms
Cannot be abstracted
Platform developers don't want abstraction.
Platform developers don't *want* need abstraction
Useful for high-level APIs
Reduce fragmentation
No need to be familiar with different toolkits
Improve teamwork
Empower developers
Give more choice
Simplify
Qt as a platform
Qt IS a platform
Qt

Code less.
Create more.
Deploy everywhere.
Deploy everywhere?
tick
but
It's not always convenient
Deploy everywhere,
Support all that's needed
Thanks!

Questions?

Dario Freddi
dario@ispirata.com