

Project Thesis

Extending the Window Manager for Novel Interaction Techniques

Alexander Lang

langal@cip.ifi.lmu.de

LMU Media Informatics Group

22.06.10

Supervisor: Raphael Wimmer

Professor in charge: Prof. Dr. Heinrich Hußmann

- 1 Motivation
 - Interaction techniques
- 2 Related Work
- 3 The Linux Desktop
 - X11
 - Compiz
- 4 Plugins for Compiz 0.9
 - D-Buffer
 - Bullet
 - FBO
- 5 Future Work
 - Improvements
- 6 Demo
- 7 References
- 8 Code Repositories

Goals:

- Evaluate new interaction techniques
- Study of technical feasibility
- Focus on customization of the workspace
- Use existing windowing infrastructure

The Physical Desktop

A Document-Centric Design

Lowfat by Mirco Müller



fps: 30
Loaded Images: 7
fpsCount length: 30

Name: Maraetaibeforesunrise.jpg
Size: (1920, 1278)
Date: 14-apr-2010 22:53



The Physical Desktop

A Document-Centric Design

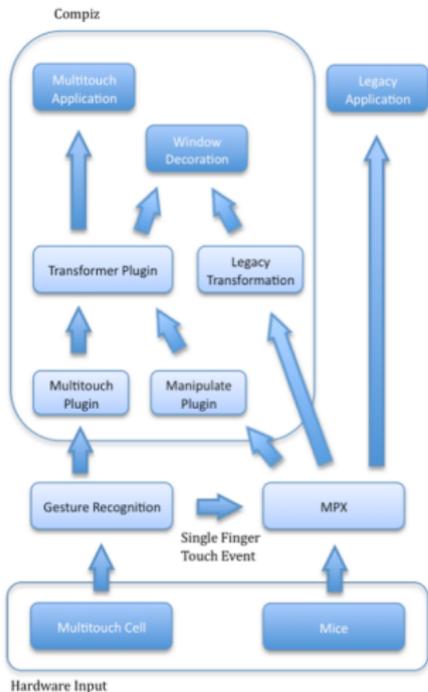
BumpTop



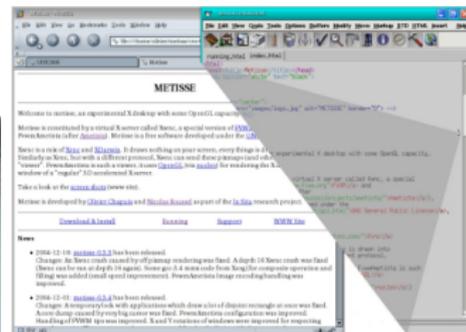
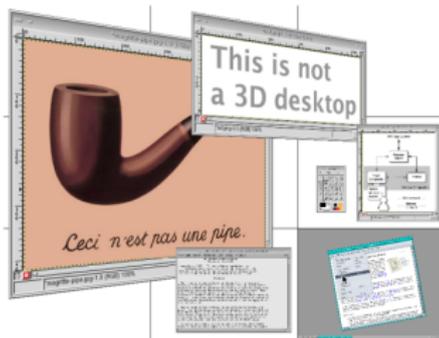
http://blog.freeware.de/uploads/bumptop_screenshot_with_printer.jpg
From "Trends and Evolution of Window Interfaces" by Martin Tomitsch

Interaction frameworks

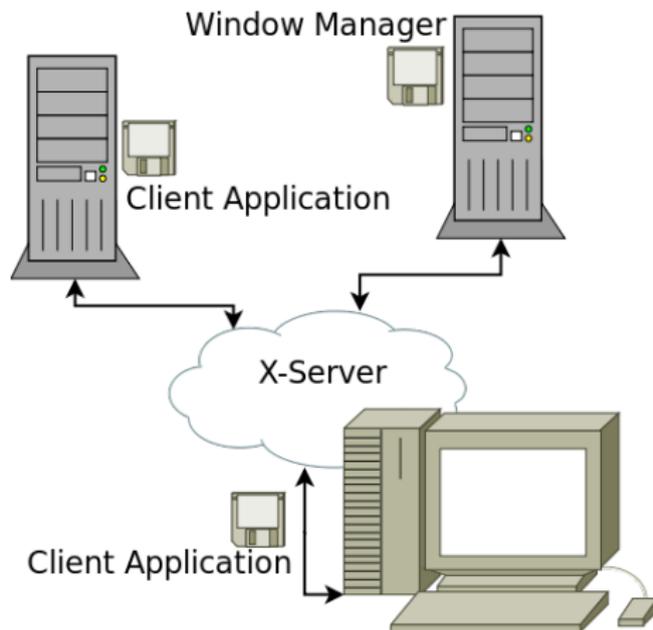
Kelvin Cheng et al. *A unified multi-touch & multi-pointer software architecture for supporting collocated work on the desktop*



Chapuis O., Roussel N.
Metisse



- Client-Server-Architecture
- X-Server manages keyboard, mouse and display
- X-Server creates windows
- “Mechanism instead of Policy”
- Window manager draws frames around top-level windows
- Window manager is just another client and can be replaced at runtime



Compiz

An OpenGL-Based Compositing Window Manager

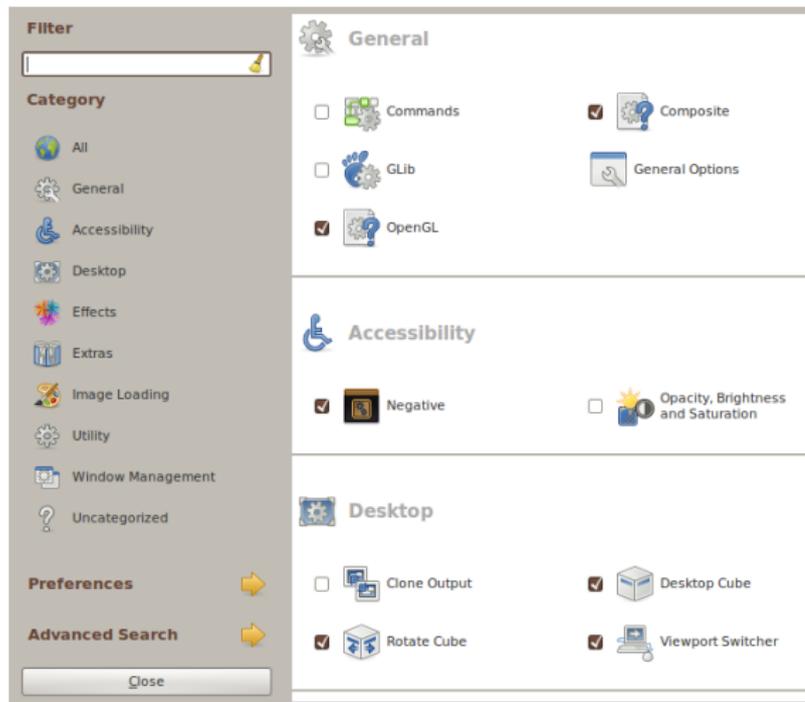
- Open source
- Originally written in C
- uses texture_from_pixmap extension
- Enabled by default in Ubuntu Linux
- First released in 2006
- Compiz 0.9 (C++ port) in development since end of 2008



- Plugins can be created by inheriting from special classes
- Plugins are compiled to a shared library by the CMake build system.

The Compiz Configuration Settings Manager (ccsm)

- Part of Compiz distribution
- Python UI
- Installed plugins can be toggled at runtime



Display-Buffer

Uses image buffers for information storage

D-Buffer plugin is based on the Buffer Framework by Isenberg et al.

Idea: Use screen-sized image buffers to save information.

Each buffer saves data for a specific effect

- Draw into the buffer, similar to any paint application
- Move windows on painted areas
- Underlying buffer changes attributes for that window e.g. size, opacity or invokes arbitrary scripts and applications

Customize your workspace

Mark regions on screen you are comfortable to work with, especially relevant to tabletops and large displays.

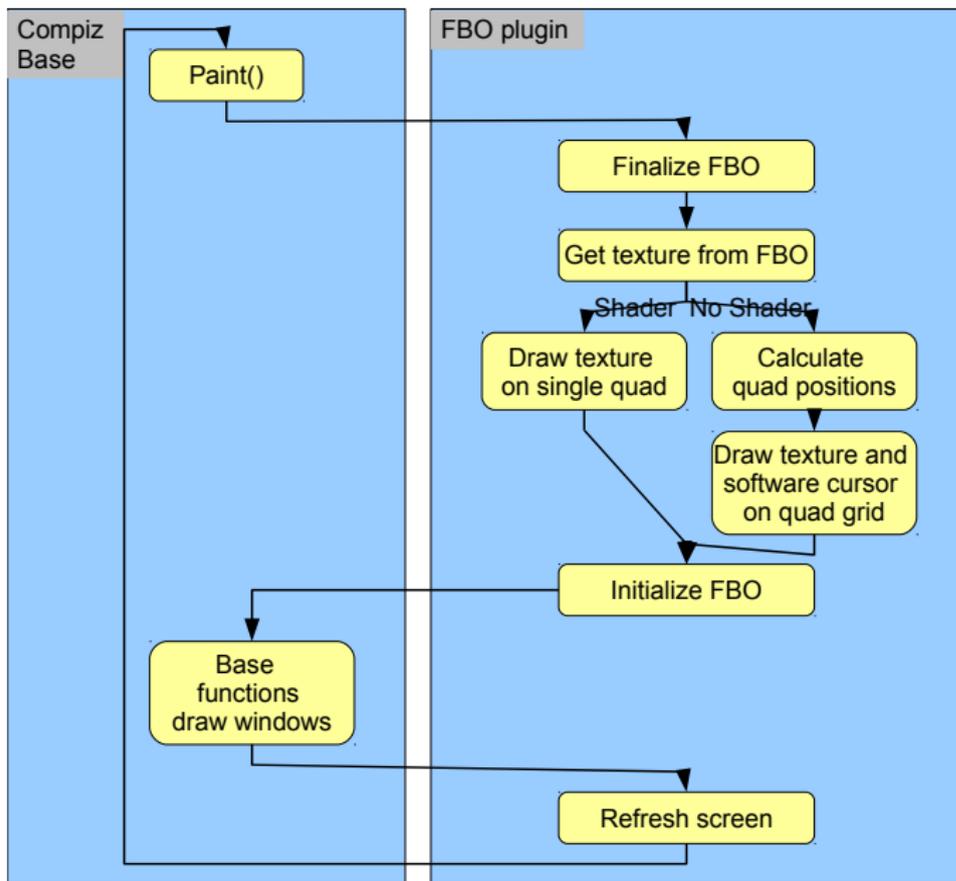
Bullet

Plugin that realizes an integration of the Bullet physics engine

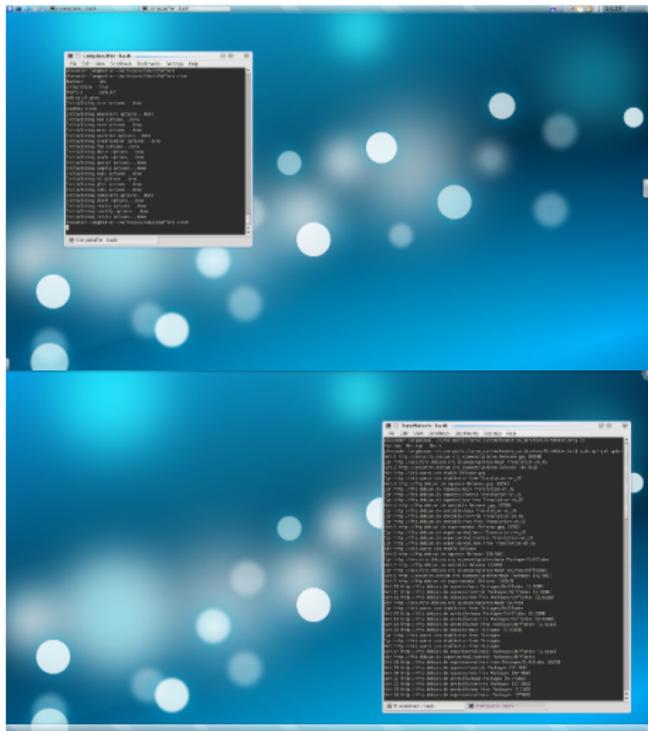
- Windows keep momentum
- Windows collide
- Border around visible area, windows cannot disappear.
- Attraction/Repulsion

FBO: Render-to-Texture via a framebuffer object

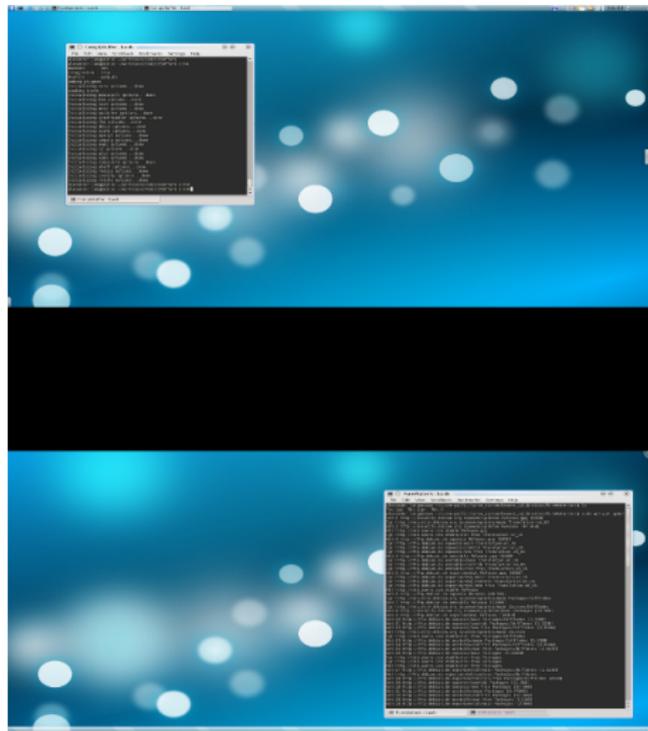
Provides (sort of) distortion correction for CURVE



CURVE: Original output and corrected output



The left column contains two screenshots of a terminal window titled "Hardware-Test". The top screenshot shows the original output, which is a list of hardware information including system name, kernel version, and various system parameters. The bottom screenshot shows the corrected output, which is a more detailed and structured list of hardware information, including system name, kernel version, and various system parameters.



The right column contains two screenshots of a terminal window titled "Hardware-Test". The top screenshot shows the original output, which is a list of hardware information including system name, kernel version, and various system parameters. The bottom screenshot shows the corrected output, which is a more detailed and structured list of hardware information, including system name, kernel version, and various system parameters.

- Optimize high usage of resources resulting from constant screen refresh
- D-Buffer as separate Application?
- Multi-touch
- Evaluate further use for buffer in a document-centric work-flow
- Further uses for physics: “Grappling Hook” to reach distant windows

Video

References

- A. Agarawala and R. Balakrishnan. Keepin'it real: pushing the desktop metaphor with physics, piles and the pen. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, page 1292. ACM, 2006.
- K. Cheng, B. Itzstein, P. Sztajer, and M. Rittenbruch. A unified multi-touch & multi-pointer software architecture for supporting collocated work on the desktop.
- T. Isenberg, A. Miede, and S. Carpendale. A buffer framework for supporting responsive interaction in information visualization interfaces. In *Creating, Connecting and Collaborating through Computing, 2006. C5'06. The Fourth International Conference on*, pages 262–269, 2006.
- O. Chapuis and N. Roussel. Metisse is not a 3D Desktop! In *Proceedings of the 18th annual ACM symposium on User interface software and technology*, page 22. ACM, 2005.
- <http://www.compiz.org>
- <http://cairographics.org/>
- <http://bulletphysics.org/wordpress/>

Code is available under

- <http://www.cip.ifi.lmu.de/~langal/git/compiz/Buffer>
- <http://www.cip.ifi.lmu.de/~langal/git/compiz/Bullet>
- <http://www.cip.ifi.lmu.de/~langal/git/compiz/FBO>

Instructions

```
git clone <url>
mkdir build
cd build
cmake ..
make
(sudo) make install
```

Compiz must be installed.

Bullet must be installed as shared library.

Plugins have to be activated in the Compiz Configuration Settings Manager (ccsm)