New Evolutions in the X Window System

Matthieu Herrb and Matthias Hopf

Introduction

- The X Window System is 20 years old

- Started new evolutions in last years

- Some new features are already available

- On-going developments to appear in X11R7

- Future evolution: XGL and XEGL.

The XFree86 → X.Org transition

1992 : The free version of X386 becomes XFree86 to avoid confusion with Thomas Roell's commercial X386.

2003 : disagreements on XFree86 management. Keith Packard gets kicked out of XFree86 core team.
Some people decide to resurrect X.Org and take back the X leadership.

February 2004 : the new version 1.1 licence in XFree86 4.4 accelerate Linux distributions and other projects moving away from XFree86.

September 2004 : release of X.Org 6.8. Hosted at freedesktop.org.

February 2005 : release of X.Org 6.8.2

November 2005 : planned release of X.Org 6.9/7.0

The Render extension

- X protocol : traditional boolean operations between source and destination bitmaps.
  Mostly pseudo-color based.

- The **Render** extension adds Porter & Duff compositing (Alpha channel).

$$C_{\text{result}} = C_{\text{under}} \cdot (1 - \alpha_{\text{over}}) + C_{\text{over}} \cdot \alpha_{\text{over}}$$

  – Enables: anti-aliasing and transparency.
  – Example application: xclock.
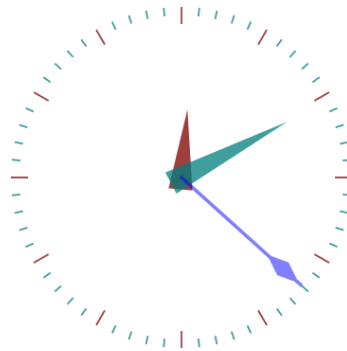
Client-side font rendering

Traditional X protocol: server-side font rendering. Has come to an end.

- applications need access to more information than just metrics

- fonts embedded in application-side documents.

Introduction of 2 libraries for client-side font rendering:

fontconfig handles font directories, maps filenames to font names, font properties and encodings. More general than X.

Xft uses fontconfig and Freetype rendering code to provide font rendering for X applications

Xft2 is adopted by most X implementations and by major toolkits (Gtk+, Qt)

Core X fonts are obsolete.

Composite and Damage extensions

Traditional X applications draw directly to the screen, don't know about other window contents.

To implement translucency:

- redirect rendering to off-screen pixmaps

- the *composite manager* manage final rendering on screen.

Also, need to notify applications of damage done on their windows by other applications: the Damage extension.

Composite Manager

Clients display directly on
the screen

Compositing
manager

Clients display off–screen
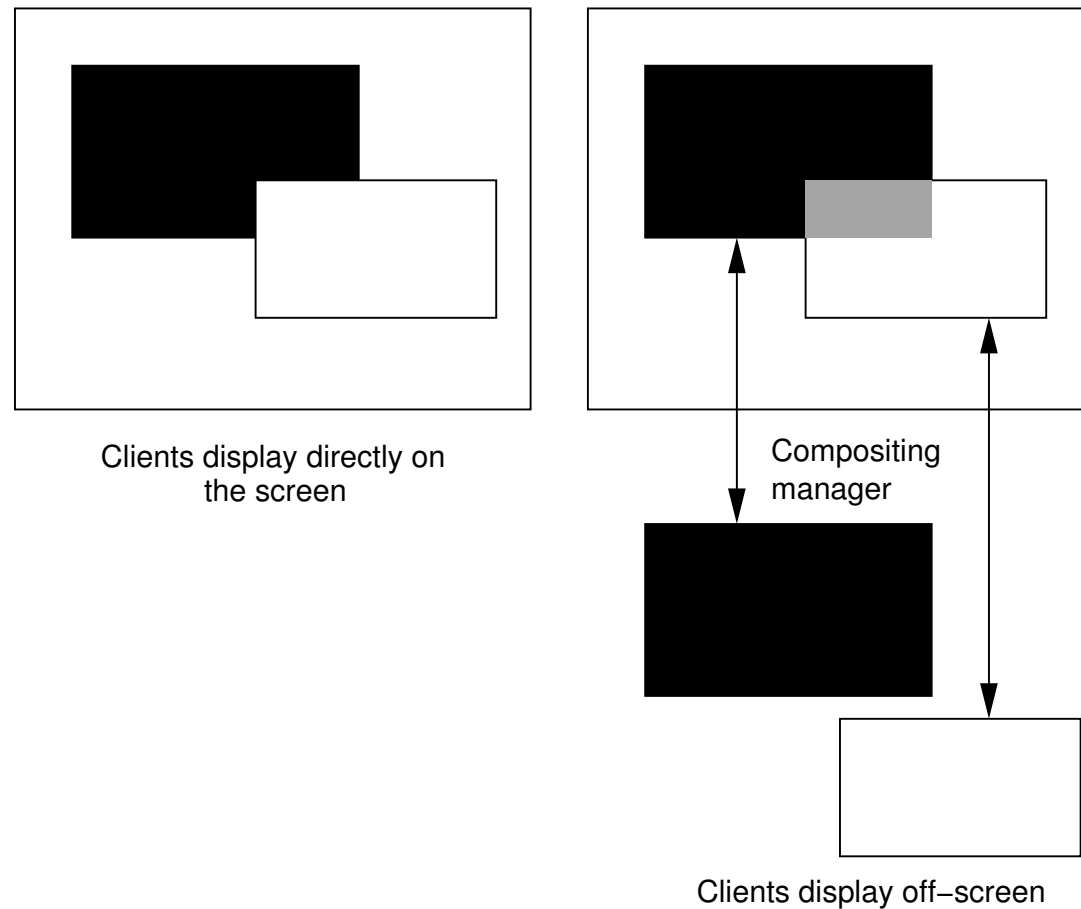
`xcompmgr` is a sample composite manager that can be used with existing window
managers.
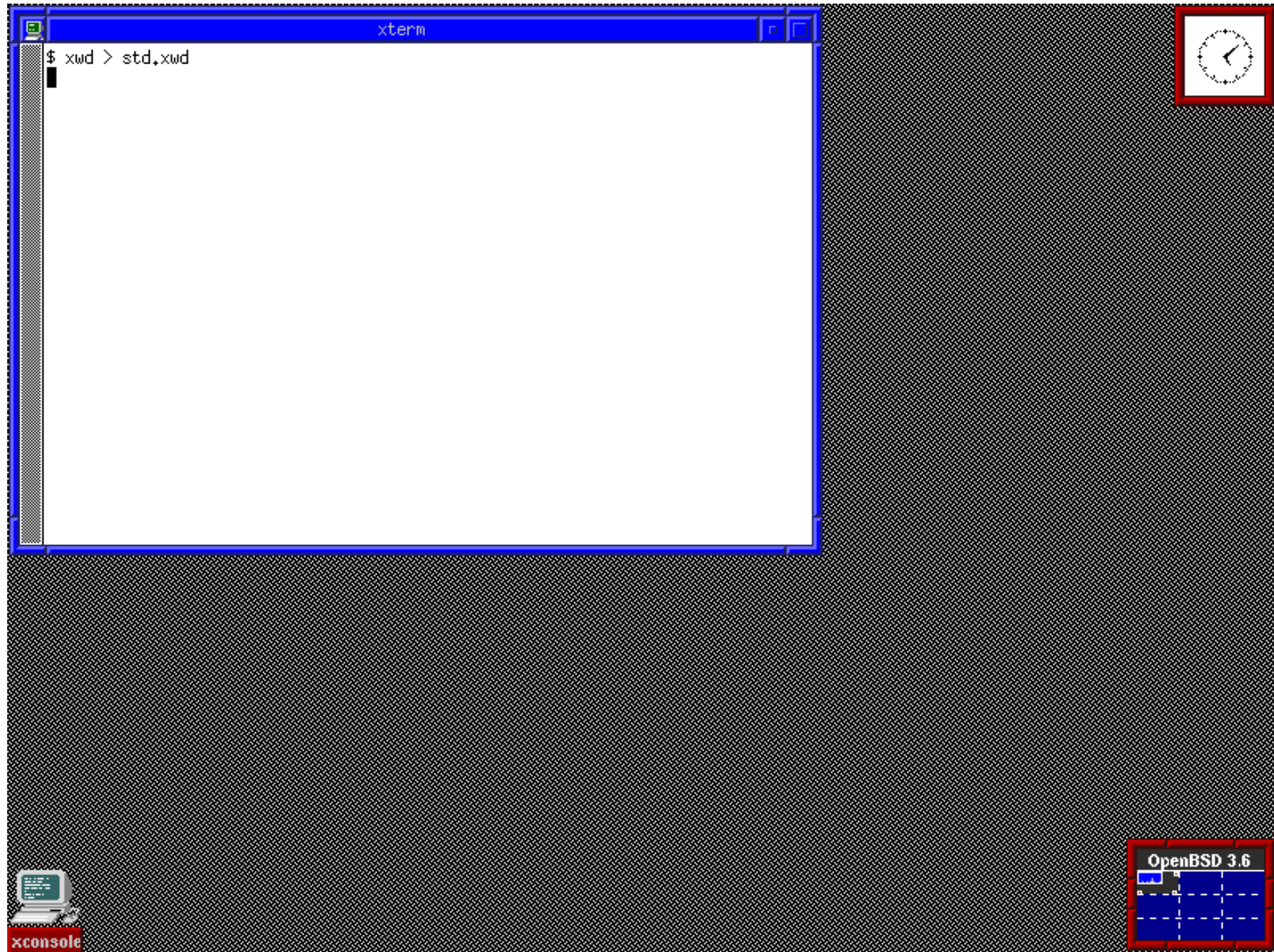
Cairo

Cairo is a 2D graphics library with support for multiple output devices. Currently supported output targets include:

- the X Window System,

- win32,

- image buffers.

Experimental backends include:

- OpenGL (through Glitz),

- Quartz,

- Postscript and PDF file output.

Cairo (2)

Cairo is designed to produce consistent output on all output media while taking advantage of display hardware acceleration when available (eg. through the X Render Extension).

The Cairo API provides operations similar to the drawing operators of Postscript and PDF. Operations in Cairo include:

- stroking and filling cubic Bézier splines,

- transforming and compositing translucent images,

- anti-aliased text rendering.

All drawing operations can be transformed by any affine transformation (scale, rotation, shear, etc.)

X11R6.9/X11R7.0

Release due real soon now.

Twin releases:

X11R6.9 last monolithic release (built with imake)

X11R7.0 first modular release (built with GNU autotools).

Includes:

- New acceleration architecture (EXA)

- Device driver updates

- Mesa 6.4 and corresponding DRI

- Better Render and Composite acceleration

A new acceleration architecture: EXA

- Render and Composite need acceleration !

- Old XAA framework has become unsuitable.
  Handles acceleration of many barely used primitives. Costs > Benefits

- Composite and Render rely mostly on off-screen rendering and need good off-screen memory management and backing store coherency.

- EXA (EXcellent Architecture) – Zack Rusin :

  - concentrates on solid fills, blits and Porter&Duff compositing,
  - provides an efficient off-screen memory manager.

- The i128, radeon and sis drivers have already been converted.

<div align="center">

`http://wiki.x.org/wiki/ExaStatus`

</div>

Driver updates

radeon

- RageTheatre 200 support
- PCI-E cards support

nv

- GeForce 6800 XT and 7800 support

i810

- Better dual-screen support
- Intel i945G and E7221 support

via

- better Xv/XvMC support

Source tree modularization

- Imake configuration has become a real mess.

- X.Org wanted a more modular organisation of the source tree: independent releases of video drivers, libraries or extensions.

- It was decided to split the tree and use autotools to manage the build of individual components.

- Autotools are supported by Someone Else.

- Appropriate for many tiny modules model.

- Work done during 2005, on the base of some previous experiments by Daniel Stone (Debrix).

- 7.0 will be supported on only a subset of supported operating systems. 7.1 will support more systems.

```
http://wiki.x.org/wiki/ModularizationWorkingGroup
```

Source tree modularization (2)

Several packages:

xproto	all the header files describing the actual X protocol and extensions. One package for the core X protocol and one package per X extension (Shape, MIT-SHM, Render, etc.),
libs	all the libraries, one package per library (X11, Xext, Xrender, etc.),
data	data files (bitmaps and icons, XKB data files, X cursors),
apps	sample applications provided by X.Org (twm, xcalc, xedit, xlogo, xman, xwd, etc),
xserver	X servers (Xorg, Xnest, Xprint, Xvfb),
drivers	graphics cards and input drivers, each one in an independent package,
fonts	several fonts packages,
doc	existing documentation
utils	various utilities that help the modular infrastructure

The future

- 2D acceleration is getting less and less attention from graphics chips manufacturers.
- The X server accessing the hardware directly is causing problems.
- New architecture, based on OpenGL, moves the drivers out of the X server.
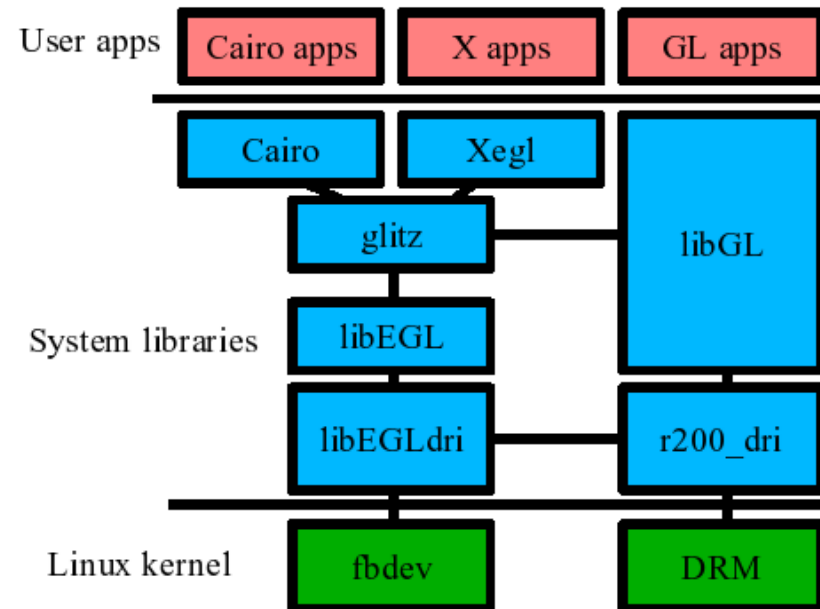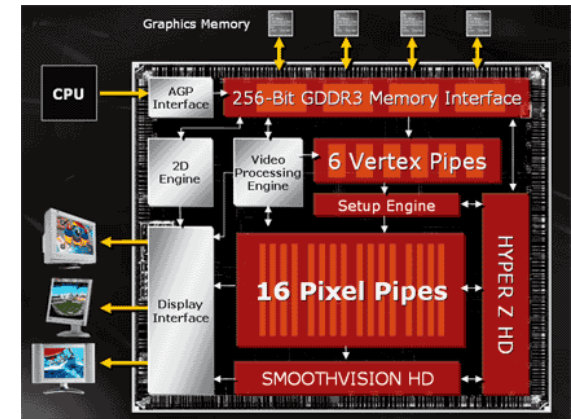
Conclusion

- X development has resumed

- provides enough features for modern desktop environments

- moving to OpenGL based acceleration

- challenges remain for Open Source systems

Support the X.Org foundation, not only your favourite BSD !