Release Notes for XFree86® 4.6.0

The XFree86 Project, Inc

April 2006

Abstract

This document contains information about the various features and their current status in the XFree86 4.6.0 release.

1. Introduction to the 4.x Release Series

XFree86 4.0 was the first official release of the XFree86 4 series. The current release (4.6.0) is the latest in that series. The XFree86 4.x series represents a significant redesign of the XFree86 X server, with a strong focus on modularity and configurability.

2. Configuration: a Quick Synopsis

Automatic configuration was introduced with XFree86 4.4.0 which makes it possible to start XFree86 without first creating a configuration file. This has been further improved in XFree86 4.5.0 and 4.6.0. If you experienced any problems with automatic configuration in a previous release, it is worth trying it again with this release.

While the initial automatic configuration support was originally targeted ust for Linux and the FreeBSD variants, as of 4.5.0 it also includes Solaris, NetBSD and OpenBSD support. Full support for automatic configuration is planned for other platforms in future releases.

If you are running Linux, FreeBSD, NetBSD, OpenBSD, or Solaris, try Auto Configuration by running:

```
XFree86 -autoconfig
```

If you want to customise some things afterwards, you can cut and paste the automatically generated configuration from the `/var/log/XFree86.0.log` file into an `XF86Config` file and make your customisations there. If you need to customise some parts of the configuration while leaving others to be automatically detected, you can combine a partial static configuration with the automatically detected one by running:

```
XFree86 -appendauto
```

If you using a platform that is not currently supported, then you must try one of the older methods for getting started like `xf86cfg`", which is our graphical configuration tool. It also has a text mode interface that an be used to create an initial configuration file and customise existing configurations.

After that, you can use XFree86 server's ability to create a starting configuration file. To do so you must login as root, and run:

```
XFree86 -configure
```

and then follow the instructions.

Finally, if all else fails, the trusty old standby text-based tool "`xf86config`" can also be used for generating X server config files.

Hopefully, at least one, and perhaps all of these various configuration options will give you a reasonable starting point for a suitable configuration file but we think that with the automatic mechanism you will probably not need any.

If you would like to customise the configuration file, see the XF86Config manual page. You should also check the driver-specific manual pages and the related documentation, which is found at *tables below* (section 4., page 15).

Before downloading any of the binary distributions for this release, read through the Installation Document as it can point out which particular binary you should download.

The next section describes what is **new** in the latest version (4.6.0) compared with the previous full release (4.5.0). There are many new features in this release and we unfortunately do not have enough space to cover them all here.

3. Summary of new features in 4.6.0.

This is a sampling of the new features in XFree86 4.6.0. A more complete list of changes can be found in the CHANGELOG that is part of the XFree86 source tree. It can also be viewed online at our CVSweb server `<URL:http://cvsweb.xfree86.org/cvsweb/xc/programs/Xserver/hw/xfree86/CHANGELOG?rev=HEAD>`.

3.1 Security Updates

The following security issue has been fixed in this release:

- X server pixmap allocation flaw (CERT Vulnerability Note VU#102441 `<URL:http://www.kb.cert.org/vuls/id/102441>`, CAN-2005-2495 `<URL:http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2005-2495>`).

3.2 Video Driver Enhancements

- Remove cfb support from the i740, nsc, s3virge, siliconmotion and via drivers.

3.2.1 aspeed

- A new driver for ASPEED Technologies' AST2000 graphics chips.

3.2.2 ati

- Reinstate support for all ATI adapters on all architectures supported by the XFree86 common layer.
- Restore accuracy of atimisc's clock probe despite certain I/O errors.
- Change the atimisc driver on big-endian systems to allow depth 24 even if the big-endian aperture is not available, by swapping RGB masks.
- Fix an ATI driver probe crash.
- Workaround for screen-to-screen copy corruption in second-generation integrated controllers.
- Improve accuracy of atmisc's calculations that are based on a 29.5 MHz reference clock.

- Chrontel 8398 related changes to the Mach64 support.
- Fix one cause of video memory corruption when using XAA with second generation integrated Mach64 controllers.

3.2.3 glint

- Fix the size of the DAC registers storage for pm2v and pm3 chipsets.

3.2.4 i740

- Save and restore more of the mode on server entry.

3.2.5 i810

- Make sure the DDC-probed parameters are applied before modes are selected.
- Fix some bad alignment problems when allocating memory from the end of AGP space.
- Adjust the display width according to the VESA BIOS scanline requirements.
- Add support for the Intel E7221 server chipset.
- Improve display switching support for mobile chipsets using the Fn+F? combination keys.
- Add DirectColor support for the i830 and later.
- Add gamma correction support for the i830 and later.
- Add rotation support for the i830 and later.
- Improve refresh rate selection for custom modelines.
- Add support for the 945G and 945GM.
- Support custom video modes in the video BIOS, when available.
- Fix some offset, pitch, and overlay scaler size problems with the video overlay. This solves various spontaneous lockups.
- Fix text restoration when display switching has been activated.
- Add support to check the built-in VESA mode list.
- Fix XVideo when operating in one line mode and video is pushed off the top of the screen, and fix some XVideo skew problems at certain resolutions.
- Sync the accelerator before calling AdjustFrame to avoid lockups.
- Fix a lockup when spend and resume when playing video (shutdown the overlay on LeaveVT).
- Support 1920x1088 for HDTV movies using a new option (LinearAlloc) to add more memory for the video overlay.
- Add ACPI hotkey switching support.

3.2.6 mga

- Reduce the Mystique's SYSCLK from 197 MHz to 166 MHz, allowing the higher clock to still be used when the "OverclockMem" option is activated. This fixes the instability seen on some Mystique cards, and removes the need for the workaround in previous releases.
- Revert the HALlib MGASetDisplayStart interface back to HALSetDisplayStart.
- Make 8+24 mode honour the default visual.

3.2.7 nv

- Some of the panel DPMS support in 4.5.0 only works for TMDS, and caused some panel blooming on some LVDS panels, so skip the DPMS panel code when the panel is LVDS.
- Add DPMS support for GeForce4 and newer laptops.
- Fix some console restore code for the NV11.
- Workaround for 512MB video cards (can only map 256MB of it).
- Fix a graphics acceleration hang that occurred on some NVIDIA GeForce 6xxx cards.
- Fix a potential problem with pixmap cache corruption on GeForce 6xxx and 7xxx parts.
- Add support for new hardware.

3.2.8 siliconmotion

- Merge in support for SiliconMotion's SM501 controller.

3.2.9 suncg6

- Add screen-to-screen copy, solid fill, and CPU-to-screen colour expansion acceleration.
- Map more of a Sun CG6's video memory for use by the pixmap cache.

3.2.10 sunffb

- Prevent turning off the FPU on NetBSD.
- Fix build problems on NetBSD (global register usage declaration, and increase stack usage).
- Workaround for VRAM corruption when unblanking FFB1 adapters.

3.2.11 tdfx

- Add DDC2 support.

3.2.12 trident

- Do not wait for vertical retrace on PC98 systems.
- Fix an offset problem in Xv for the Cyber9397 and later chipsets.
- Add XGI (Trident) Volari XP5 support to the trident driver.

3.2.13 via

- Add a "ConnectedDevice" option to the via driver to force selection of output devices.

3.2.14 vmware

- Implement a workaround in the generic int10 module to allow unaligned I/O access specific to VMWare's second generation emulated adapter.

3.2.15 xgi

- A new driver for the Volari V3XT, V5, V8, Z7 chipsets.

3.3 Input Driver Enhancements

3.3.1 keyboard

- Ignore `KEY_UNUSED` scancodes.

3.3.2 mouse

- Fix auto-detection of Sun mice on Linux/SPARC.

3.3.3 magictouch

- Port of the magictouch driver to XFree86 4.x.

3.4 XKB updates

- Replace the macintosh/fr map with the "fr_new" map, release 4.
- Fix a bug in XKB group adjustment.
- Fix off-by-one errors in XKBMalloc.c.
- Shrink tables instead of only growing them.
- Fix various vmodmap vs modmap cut and paste bugs.
- Improve synthesized core X events for reporting XKB changes to XKB-unaware clients.
- Fix a setxkbmap segfault.
- Add Eszett (ssharp) to the us_intl map.
- Add dvorak_intl map.
- Add a Khmer layout.
- Add multimedia keysyms for the Inspiron 6000 laptop to the "inspiron" set.

3.5 X Server and Extension Updates

- Updates to the handling of the X server RGB database:
    - Ignore white space, dash, and underscore characters, as well as case when matching colour names, and remove the unnecessary duplicates in rgb.txt.
    - Allow a copy of the RGB database to be included into the X servers as a fallback for when the rgb.txt file cannot be found.
    - Remove the (unused) support for DBM-compiled RGB database files.
- Make the AMD64/x86_64 platform tests consistent throughout.
- Rework the way that extension support is built into the servers to allow for servers with different sets of extensions in the same build. This includes minimising the number of pervasive extensions, and adding DIX helpers for extensions such as Xinerama and XCSECURITY.
- Modify the X server handling of SIGINT to provide a faster exit path than SIGTERM, while ensuring that all hardware state gets restored. This change is used by xinit as a cleaner alternative to using SIGKILL for a slowly exiting server.

3.5.1 Loader

- Add support for relocation type 10 for Alpha to the ELF loader.
- Fix issues with Linux/SPARC 2.6 kernels.
- Add loader infrastructure to implement inter-module version checks.

- Use mprotect on NetBSD.
- Fix support for NetBSD/arm.
- Add support for `R_SPARC_OLO10` relocations.
- Make error messages for unsupported relocations more useful.
- Improve the loader server's emulation of shmctl() calls.
- Fix some problems with the a.out header macros used by the loader, which was causing incorrect header interpretation for some a.out object types.
- Major loader enhancements, including:

  - Per-module export symbol lists.
  - Improved symbol hash.
  - A mechanism for specifying the scope of symbols.
  - Keep track of relocations, and undo them when they become unresolved on module unload, and recalculate them when they change on module load.
  - Check relocations on each module load/unload, ensuring that function references that are unresolved at any time are always mapped to the catch-all function, preventing calls into nowhere.
  - Efficient handling of invariant relocations.
  - Cleanup/update the memleak stack trace code, and print a stack trace when an unresolved function is called or when XFree86 aborts.
  - Update the catch-all function to identify the unresolved function that was called.
  - Read the XFree86 executable's symbol table in order to get the most useful stack traces, and use dladdr() where available for identifying shared library references.
  - Cleanup and unify debug messages for the loader components.
  - Use module-instance-specific referenced and required symbol lists.
  - Update drivers to make use of some of the new loader functionality, including the instance-specific ref/req lists, and some unloading of modules when they are no longer needed.
  - Don't build the coff loader on platforms that it doesn't support.
  - Use stronger typing of the module-visible instance-specific module descriptor.
  - No longer disable xf86UnloadSubModule() since the loader now handles unloads and reloads cleanly.
  - Do not defer unloading drivers when using autoconfig, since that is not necessary with the loader updates.

3.5.2 XFree86 core server and modules

- Modify the handling of preferred EDID modes to allow user-specified modes that are larger than the preferred mode.
- Refine VGA routing emulation through Sun's Simba P2P bridges.

- Allow a static server's -configure to generate an XF86Config for the loader server.
- Fix a bug relating to the forwarding of 64-bit memory addresses by PCI-to-PCI bridges on 32-bit systems.
- Fix a bug in the decoding of I/O forwarding by Cardbus bridges.
- Update -configure to detect when wscons support should be used for the keyboard on NetBSD.
- Fix an endianness bug in the xf4bpp module.
- Fix some of the endianness bugs in the vbe module.
- Remove obsolete VBE functions that have not been compiled-in for some time.
- Fix VBE's reporting of video memory size.
- Re-implement Alpha kludge in x86emu's ops.c to be more portable.
- Intercept "raw" requests for PCI configuration space from video BIOSes on all architectures.
- Handle unaligned I/O requests more gracefully and be more tolerant of spurious I/O errors in the int10 module.
- Make sparse I/O masks more portable throughout the server.
- Fix a long-standing off-by-one bug in the setting of RGB offsets when the driver supplies RGB masks.
- Make I/O error handling more reliable on SPARC.
- Change the OS-specific module subdirectory name for all SunOS variants from "solaris" to "sunos".
- Port of SBUS drivers to SunOS variants. This also allows for multihead using a mix of SBUS and PCI devices.
- Fix text mode restore on x86/Solaris 8 and later.
- Change autoconfiguration to include PCI and SBUS BusIDs in generated device sections. This fixes some problems when fbdev is active.
- Prevent the sharing of a PCI entity by more than one driver.
- Insulate various common layer functions against out-of-bounds entity indices.
- Fix XAA's support for `TRIPLE_BITS_24BPP` on big-endian systems.
- Fix unaligned access in XAA's handling of 24bpp.
- The xf24_32bpp module is no longer used by any of our drivers, so don't include it in static builds.
- The xf8_32wid module is only used by the sunffb driver, so don't include it in non-SPARC static builds.
- Fix generic int10 memory accesses on little-endian systems.
- Fix a SEGV that can occur under certain builds of glibc 2.3.4 and up, when linux/int10 attempts to determine kernel support for the vm86() system call.
- Add an xf86DeallocateGARTMemory() function.
- Make autoconfig stop at the first successful probe on each retry, minimising unnecessary probes.

3.5.3 TinyX

- Allow build-time selection of which font renderers get built in to the TinyX servers.
- Fix a problem where VT switching back to TinyX servers was failing due to blocking on keyboard read.
- Fix a TinyX build issue under `!XserverStaticFontLib`.
- Move Xserver/*/tiny directories under Xserver/hw/tinyx.
- TinyX build fixes.

3.5.4 Xnest

- Add a "-noinput" option that allows Xnest to be run in a view-only mode.

3.5.5 Xsun

- Fix incorrect bitsPerPixel for depth 12 in pixmap formats advertised by the Xsun24 server.

3.5.6 cfb

- Fix a bug that occurs when using cfb to tile 32-bit rectangles on SPARC.
- cfb24 is no longer needed. Keep the loadable module, but no longer statically link it into servers.

3.5.7 shadow

- Rewrite shadowUpdateRotatePacked() to reduce cache misses.

3.5.8 Xinerama

- Cause the server to FatalError() when Xinerama attempts to create a screen whose root depth has no visuals, causing Xlib to segfault.
- Allow Xinerama to initialise even when there is only one screen to combine.
- Make loadable modules and MI independent of Xinerama:

    - Add a DIX helper to determine whether Xinerama is active and use it instead of #ifdef's code in all loadable modules.
    - Re-implement, as a screen wrapper, Xinerama's PaintWindow*() hacks.
    - Move down to DIX Xinerama's only MI hook.

3.5.9 XFree86-DGA

- Do not call a screen's PointerMoved entry if a DGA mode is active on that screen.
- Fix client-side framebuffer mapping on NetBSD.
- Make sure that DGAShutdown() is always called before HW state is restored on exit and reset, and make sure that it is not called after the HW state has been restored.

3.5.10 Render

- Fix unaligned access in RenderCompositeGlyphs request.

3.5.11 RandR

- Use xf86SetViewport as well as SetCursorPosition during RandR rotation, as xf86SetViewport will call AdjustFrame to reposition the window if necessary and avoid the cursor ending up off screen.

- Add a new function xf86GetRotation() to allow third party modules to obtain the current rotation.
- Fix up error case when rotation fails and the framebuffer needs to be re-enabled.

3.6 Library, Client and Utility Updates

3.6.1 Xlib

- Fix typos in XErrorDB that were preventing descriptive messages for some extension errors.

3.6.2 Xt

- Fix an uninitialised pointer dereference in Xt/Initialize.c on WIN32 platforms that happens if `$HOME` is set in the environment.
- Add support to Xt's string to float converter to read resources in fractional notation.
- Add some XChar2b string manipulation functions to Xt.
- Bump the minor revision of the Xt shared library to reflect the fact that some new interfaces have been added.

3.6.3 Xaw

- Extend the tool tip resource for Xaw's simpleWidgetClass to allow a function to be called
- Fix problems with Xaw's label widget when using XChar2b strings.
- Add a new connectionType property to Xaw's tree widget.
- Make Xaw Form widgets inherit dimensions from their parent at widget initialisation time.
- Add mouse wheel support for Scrollbars in Xaw's Viewport widget.
- Fix a performance issue with Xaw's Tree widget caused by useless relayouts.
- Fix i18n for Xaw's tooltip widget.
- Add two new functions that can be used to scroll the content of an Xaw viewport from outside the viewportWidgetClass.
- Fix the cursor in Xaw textWidget when the input field has focus.
- Bump the minor revision of the Xaw shared library to reflect the fact that some new interfaces have been added.
- Add user_data field to Xaw's simpleWidgetClass.
- Fix Xaw viewport scrollbar flickering problem.

3.6.4 pswrap

- Fix a problem where pswrap generates a bogus .c file when the -h option is not used.
- Document pswrap's -f option.

3.6.5 mmapr, mmapw

- Extend mmapr and mmapw on Linux to allow specification of which PCI address space (I/O or memory) to mmap.
- Add options to mmapr and mmapw to allow unaligned accesses for testing purposes

3.6.6 cpconfig

- Use the same default XF86Config search path as the XFree86 server.

3.6.7 xterm

These changes correspond to xterm patches #200 through #212.

- Improved behavior

    - add a new selection feature: regular expressions, and new resources which specify what happens on multiple mouse clicks: on2Clicks, on3Clicks, on4Clicks and on5Clicks.
    - improve initialization of toolbar so that individual pulldown menus are initialized on demand, as they are in the scenario where they are invoked as popup menus.
    - improve error-reporting when chown/chmod of the pseudo-terminal fails, e.g., if a copy of xterm which was designed to work with old-style pseudo-terminals is not installed setuid or setgid.
    - add support for interpreting the underline attribute as an italic font in Xft mode.
    - modify configure script --disable-imake to use the script's definitions anyway if it cannot detect imake.
    - improve resource files to show how the menubar and popup menus can be colored.
    - modify initialization of allowSendEvents and allowWindowOps to prevent modification with the editres protocol.
    - improved some of the built-in line-drawing glyphs.
    - set the _NET_WM_PID property.
    - update table for mk_width() from UnicodeData 4.1.0.
    - implement the remaining pieces to make xterm allocate cells for wide-characters when the "UTF-8" menu entry is selected. Also, load the UTF-8 font when that menu entry is selected, or the escape sequence for UTF-8 mode is received :
    - improve behavior when switching to UTF-8 mode after startup so xterm will check if the current fonts are already wide (ISO-10646-1). If they are not, xterm will use the utf8Fonts subresource to load appropriate fonts.
    - modify initialization for wide-bold fontname, to search for one if none is given.
    - modify interaction between +u8 and locale resource to allow the command-line option to override the resource.
    - modify initialization to decide whether to default to built-in wcwidth() versus system's version based on the starting locale and whether the system's version is poor quality.
    - make paste of UTF-8 faster for Western character sets by checking range of incoming data.
    - improve initialization due to utf8 resource by loading the utf8Fonts resource in the case where locale resource is false. Also in this case, do not disable switching UTF-8 mode on/off.

- Modified behavior

  - modify parsing of OSC (and SOS, etc), strings so their contents are not interpreted as UTF-8. This allows non-ASCII title strings to be set, provided that the window manager complies.
  - modify parsing of control sequence CSI T to allow scroll-down to be sent while mouse tracking is enabled.
  - modify command-processing to accept an optional parameter that tells xterm which shell program to use.
  - change xterm manpage to show the actual color resource names XtDefaultForeground and XtDefaultBackground rather than black and white.
  - modify xterm-new terminfo entry to use capabilities for shifted scroll forward/reverse as shifted cursor up/down.
  - remove menu entry for "Enable Curses Emulation" (to make room for "Select To Clipboard").
  - modify resource files to make the font-resources a little more specific, e.g., changing "*VT100*" to "*VT100.", to make the distinction between VT100.font and VT100.utf8Fonts.font sharper, in case a packager modifies one of those.
  - allow cursor to have the same color as foreground (text), since it is rendered as reverse.
  - modify Imakefile to use setgid mode for installing with Linux, OpenBSD and FreeBSD.

- New resource settings

  - add utf8Title resource and menu entry, allowing the user to control whether title strings are interpreted as ISO-8859-1 or UTF-8 encoding.
  - add scrollBarBorder resource.
  - add resource selectToClipboard, action set-select and a menu entry to allow users to switch between PRIMARY and CLIPBOARD for select/paste.
  - add resource mkWidth and command-line option -mk_width to control whether xterm uses the built-in version of wcwidth().
  - add resource settings for minimum/maximum input buffer size, and call to sched_yield to improve performance with newer Linux kernels
  - add simpler resource keyboardType which, when set, overrides the individual keyboard-type resources and eliminates the possibility of conflict between them.

- New configure script options

  - the --with-utmp-setgid allows the packager to configure xterm to use setgid permissions for utmp.
  - add configure option --enable-readline-mouse, which turns on the experimental OPT_READLINE code.

- add --enable-narrowproto configure option to accommodate X.org "modular" build.
- add configure --with-app-defaults option to allow app-defaults directory for install-rules to be customized.
- add configure option --with-tty-group to help work around deficiencies in packager's build environment.
- add configure --with-setuid and --with-reference options to allow packagers more flexibility in customizing install permissions.
- add configure --with-symlink option to make it simpler to install successive versions of xterm renamed using the --program-prefix, etc., options with a symbolic link pointing to the most recent.
- add a configure check for the actual path of luit, to work around broken imake configurations.
- add a configure check for pkg-config.
- modify install rule for uxterm to account for the --program-prefix, etc., options by invoking the transformed name of xterm rather than simply "xterm".
- modify configure --with-app-defaults option to allow --without-app-defaults.
- add configure option --disable-leaks.
- add experimental option to allow applications to get or set the selection data as a BASE64 string.

- Other new features
  - add select-cursor-extend() action.
  - add environment variables $XTERM_SHELL and $XTERM_VERSION.
  - add menu entry (alt-esc) and corresponding action (alt-sends-esc) to toggle the eightBitInput resource setting.
  - add command-line options (-tb, +tb) and resource toolBar to allow menu/toolbar to disabled or enabled at startup.
  - add extended shift- and control-modifier cursor keys to "xterm+pcfkeys" terminfo entry to correspond to ncurses 20050430 patch.
  - enable lastlogx support for NetBSD (was added, but not enabled in patch #186).

- Bug-fixes (see http://invisible-island.net/xterm/xterm.log.html for a more complete list):
  - correct typo in configure script's --enable-dec-locator option
  - correct a typo in CF_FUNC_TGETENT introduced in patch #198 fix
  - fix an off-by-one error parsing -S/nn option.
  - add a check in Bell to ensure that the VT100 widget is realized, since it may be called by xtermLoadFont given an incorrect font resource.
  - correct treatment of iconBorderWidth for resizing an active-icon, and its description in manpage.

- add a limit check for scrolling margins in a one-line screen, overlooked in fixes for patch #198.
- correct computation of width for wide characters with the invisible attribute.
- improve fix from patch #198 for Cleanup() by ensuring it is not called from the SIGCHLD handler.
- fixes to make -geom option work properly with the toolbar configuration.
- fix logic in find_utmp, which did not reset result in getutid(), causing an infinite loop in some conditions.
- fix a rare case where text would be written with the wrong colors because output of scrolled text would reset the colors and the new text would be written with the same colors.
- implement logic in termcap query to process multiple parameters as documented in ctlseqs.ms
- fix buffer size used for termcap query, which was not long enough for the terminfo "colors" name.
- initialize the saved-cursor data so a restore-cursor operation without a preceding save-cursor operation will not modify the foreground color.
- modify Imakefile to work around old problems in imake configuration to allow test-builds using xmkmf on Linux. (This was not noticed since several releases had broken definitions relating to Xft which were harder to work around).
- remove default translations for dabbrev-expand() due to conflicts with existing keyboard arrangements.
- set icon border width explicitly to work around fvwm problem with active icon resizing.
- eliminate a retry for a better-matching bold font, to work around recent font server changes.
- modify prefix/suffix transformation in makefile to make uxterm install properly on Cygwin.
- correct termcap "me" (mode-end) string so it does not modify the alternate character set.
- modify computation of rows/columns on resize to avoid extending beyond the given limits, e.g., if resizing in response to a "maximize" in Gnome or KDE which do not use the window manager hints for this case.
- make a special case of resizing work like vt100: a hard reset also resets the 132/80 mode. The code to support this was present since X11R5, but not used because the corresponding initial state of the -132 option was not saved.

3.6.8 GLX/DRI

- Mesa build fixes for NetBSD/sparc64.
- Solaris/SPARC 64-bit libGL fixes.

3.6.9 dbedizzy

- Resurrect this DBE extension test utility.

3.6.10 xclock

- Use the Xaw tooltip to display the date in xclock.

3.6.11 twm

- Add a new function to allow the window text title to be changed interactively.
- Allow environment variables to be used in menu names.
- Add a new "f.totalzoom" function that zooms windows to cover the whole area of a screen with no window decoration visible.
- Don't highlight "f.nop" menu items in menus.

3.6.12 xbiff

- Add a tooltip to xbiff that shows the number of messages in a mail folder.

3.6.13 xdm

- Fix a potential infinite loop when the "willing" command fails to produce any output.

3.6.14 xfd

- Fix a bug that prevents properly paging through xft fonts beyond 0xFFFF.

3.6.15 xfs

- Modify the font server transport messages to go through xfs's logging mechanism.
- Fix a bug introduced when syslog support was added that prevents exit on fatal errors.

3.6.16 xinit

- Send SIGINT to slowing exiting X servers instead of SIGKILL. This works together with the new SIGINT handling in the X servers, and results in a clean exit. Sending SIGKILL only results in the hardware state not being restored, often requiring a reboot to recover from.

3.6.17 xmessage

- Fix a bug in the handling of escapes in button labels.

3.6.18 xvidtune

- Fix the -timeout <n> option.

3.7 I18n and Fonts

- Update mkfontscale's handling of font weights.
- Replace lib/font/fontfile/decompress.c with a version derived from BSD's src/usr.bin/compress/zopen.c.
- Modify the fontconfig library to use getpwnam/getpwuid to find the home directory when `$HOME` is not set.
- Fix a bug in fontconfig's pattern matching that can cause FcFontMatch() to fail when it shouldn't.
- Fix a double-free problem in fontconfig.
- Rework printf format specifiers used in XLFD code to make them more portable and avoid infinite loops in some glibc versions.

- Update the Serbian locale naming to reflect the country's name change from "Yugoslavia" to "Serbia and Montenegro".

3.8 OS Support Updates

- Fix some cases in the Darwin build where a variable is declared static and later as extern.
- Avoid /tmp files in NetBSD.cf's man page creation.
- Install man page source as well as formatted man pages on NetBSD.
- Fix a problem where builds would continue past errors on NetBSD.

4. Drivers

4.1 Video Drivers

XFree86 4.6.0 includes the following video drivers:

Drivers marked with (*) are present in a preliminary form in this release, but are not complete and/or stable yet.

Drivers marked with (+) are for Linux/SPARC only.

Drivers marked with (-) are for Linux/mips only.

Darwin/Mac OS X uses IOKit drivers and does not use the module loader drivers listed above. Further information can be found in README.Darwin.

4.2 Input Drivers

XFree86 4.6.0 includes the following input drivers:

Drivers marked with (*) are available for Linux only.

5. Known Problems

Currently no known problems are documented.

6. Overview of XFree86 4.x.

XFree86 4.x has a single X server binary called `XFree86`. This binary can either have one or more video and input drivers linked in statically, or more usually, dynamically, and in that manner load the video drivers, input drivers, and other modules that are needed.

XFree86 4.6.0 has X server support for most UNIX® and UNIX-like operating systems on Intel/x86 platforms, plus support for Linux and some BSD OSs on Alpha, PowerPC, IA-64, AMD64, SPARC, and Mips platforms, and for Darwin on PowerPC. Support for additional architectures and operating systems is in progress and is planned for future releases.

6.1 Loader and Modules

The XFree86 X server has a built-in run-time loader, which can load normal object files and libraries in most of the commonly used formats. The loader does not rely on an operating system's native dynamic loader support and it works on platforms that do not provide this feature. This allows for the modules to be operating system independent (although not, of course, CPU architecture independent) which means that a module compiled on Linux/x86 can be loaded by an X server running on Solaris/x86, or FreeBSD, or even OS/2.

The X server makes use of modules for video drivers, X server extensions, font rasterisers, input device drivers, framebuffer layers (like mfb, cfb, etc), and internal components used by some drivers (like XAA),

Driver Name	Description	Further Information
`apm`	Alliance Pro Motion	README.apm
`ark`	Ark Logic	
`aspeed`	ASPEED Technology	
`ati`	ATI	README.ati, README.r128, r128(4), radeon(4)
`chips`	Chips & Technologies	README.chips, chips(4)
`cirrus`	Cirrus Logic	
`cyrix` (*)	Cyrix MediaGX	README.cyrix
`fbdev`	Linux framebuffer device	fbdev(4)
`glide`	Glide2x (3Dfx)	glide(4)
`glint`	3Dlabs, TI	glint(4)
`i128`	Number Nine	README.I128, i128(4)
`i740`	Intel i740	README.i740
`i810`	Intel i8xx	README.i810, i810(4)
`imstt`	Integrated Micro Solns	
`mga`	Matrox	mga(4)
`neomagic`	NeoMagic	neomagic(4)
`newport` (-)	SGI Newport	README.newport, newport(4)
`nsc`	National Semiconductor	nsc(4)
`nv`	NVIDIA	nv(4)
`rendition`	Rendition	README.rendition, rendition(4)
`s3`	S3 (not ViRGE or Savage)	
`s3virge`	S3 ViRGE	README.s3virge, s3virge(4)
`savage`	S3 Savage	savage(4)
`siliconmotion`	Silicon Motion	siliconmotion(4)
`sis`	SiS	README.SiS, sis(4)
`sunbw2` (+)	Sun bw2	
`suncg14` (+)	Sun cg14	
`suncg3` (+)	Sun cg3	
`suncg6` (+)	Sun GX and Turbo GX	
`sunffb` (+)	Sun Creator/3D, Elite 3D	
`sunleo` (+)	Sun Leo (ZX)	
`suntcx` (+)	Sun TCX	
`tdfx`	3Dfx	tdfx(4)
`tga`	DEC TGA	README.DECtga
`trident`	Trident	trident(4)
`tseng`	Tseng Labs	
`vesa`	VESA	vesa(4)
`vga`	Generic VGA	vga(4)
`via`	VIA	via(4)
`vmware`	VMWare guest OS	vmware(4)
`xgi`	Xabre Graphics Inc	xgi(4)

The module interfaces (both API and ABI) used in this release are subject to change without notice. While we will attempt to provide backward compatibility for the module interfaces as of the 4.0 release (meaning that 4.0 modules will work with future core X server binaries), we cannot guarantee this. Compatibility in the other direction is explicitly not guaranteed because new modules may rely on interfaces added in new releases.

Note about module security

Driver Name	Description	Further Information
aiptek(*)	Aiptek USB tablet	aiptek(4)
calcomp	Calcomp	
citron	Citron	citron(4)
digitaledge	DigitalEdge	
dmc	DMC	dmc(4)
dynapro	Dynapro	
elographics	EloGraphics	
eloinput	Elo 2500U USB	eloinput(4)
fpit	Fujitsu Stylistic Tablet PCs	fpit(4)
hyperpen	Aiptek HyperPen 6000	
js_x	JamStudio pentablet	js_x(4)
kbd	generic keyboards (alternate)	kbd(4)
keyboard	generic keyboards	keyboard(4)
magictouch	MagicTouch ProE-X	magictouch(4)
microtouch	MicroTouch	
mouse	most mouse devices	mouse(4)
mutouch	MicroTouch	
palmax	Palmax PD1000/PD1100	palmax(4)
penmount	PenMount	
spaceorb	SpaceOrb	
summa	SummaGraphics	
tek4957	Tektronix 4957 tablet	tek4957(4)
ur98(*)	Union Reality UR-F98 headtracker	ur98(4)
void	dummy device	void(4)
wacom	Wacom tablets	wacom(4)

The XFree86 X server runs with root privileges, i.e. the X server loadable modules also run with these privileges. For this reason we recommend that all users be careful to only use loadable modules from reliable sources, otherwise the introduction of viruses and contaminated code can occur and wreak havoc on your system. We hope to have a mechanism for signing/verifying the modules that we provide available in a future release.

6.2 Configuration

The XFree86 server uses a configuration file as the primary mechanism for providing configuration and run-time parameters. The configuration file format is described in detail in the XF86Config(5) manual page.

The XFree86 server has support for automatically determining an initial configuration on most platforms, as well as support or generating a basic initial configuration file.

6.3 Command Line Options

Command line options can be used to override some default parameters and parameters provided in the configuration file. These command line options are described in the XFree86(1) manual page.

6.4 XAA

The XFree86 Acceleration Architecture (XAA) was completely rewritten from scratch for XFree86 4.x. Most drivers implement acceleration by making use of the XAA module.

6.5 Multi-head

Some multi-head configurations are supported in XFree86 4.x, primarily with multiple PCI/AGP cards.

One of the main problems is with drivers not sufficiently initialising cards that were not initialised at boot time. This has been improved somewhat with the INT10 support that is used by most drivers (which allows secondary card to be "soft-booted", but in some cases there are other issues that still need to be resolved. Some combinations can be made to work better by changing which card is the primary card (either by using a different PCI slot, or by changing the system BIOS's preference for the primary card).

6.6 Xinerama

Xinerama is an X server extension that allows multiple physical screens to behave as a single screen. With traditional multi-head in X11, windows cannot span or cross physical screens. Xinerama removes this limitation. Xinerama does, however, require that the physical screens all have the same root depth, so it isn't possible, for example, to use an 8-bit screen together with a 16-bit screen in Xinerama mode.

Xinerama is not enabled by default, and can be enabled with the `+xinerama` command line option for the X server.

Xinerama was included with X11R6.4. The version included in XFree86 4.x was completely rewritten for improved performance and correctness.

Known problems:

- Not all window managers are Xinerama-aware, and so some operations like window placement and resizing might not behave in an ideal way. This is an issue that needs to be dealt with in the individual window managers, and isn't specifically an XFree86 problem.

6.7 DGA version 2

DGA 2.0 is included in 4.6.0. Documentation for the client libraries can be found in the XDGA(3) man page. A good degree of backwards compatibility with version 1.0 is provided.

6.8 DDC

The VESA® Display Data Channel (DDC™) standard allows the monitor to tell the video card (or on some cases the computer directly) about itself; particularly the supported screen resolutions and refresh rates.

Partial or complete DDC support is available in most of the video drivers. DDC is enabled by default, but can be disabled with a "Device" section entry: `Option "NoDDC"`. We have support for DDC versions 1 and 2; these can be disabled independently with `Option "NoDDC1"` and `Option "NoDDC2"`.

At startup the server prints out DDC information from the display, and will use this information to set the default monitor parameters and video mode when none are provided explicitly in the configuration file.

6.8.1 Changed behavior caused by DDC.

Several drivers uses DDC information to set the screen size and pitch. This can be overridden by explicitly resetting it to the and non-DDC default value 75 with the `-dpi 75` command line option for the X server, or by specifying appropriate screen dimensions with the "DisplaySize" keyword in the "Monitor" section of the config file.

6.9 GLX and the Direct Rendering Infrastructure (DRI)

Direct rendered OpenGL® support is provided for several hardware platforms by the Direct Rendering Infrastructure (DRI), which is part of Mesa. Mesa also provides the 3D core rendering component of GLX. Further information can be found at the DRI Project's web site `<URL:http://dri.sf.net/>` and the Mesa web site `<URL:http://www.mesa3d.org>`.

6.10 XVideo Extension (Xv)

The XVideo extension is supported in XFree86 4.x. An XvQueryPortAttributes function has been added as well as support for XvImages. XvImages are XImages in alternate color spaces such as YUV and can be passed to the server through shared memory segments. This allows clients to display YUV data with high quality hardware scaling and filtering.

6.11 X Rendering Extension (Render)

The X Rendering extension provides a 2D rendering model that more closely matches application demands and hardware capabilities. It provides a rendering model derived from Plan 9 based on Porter/Duff image composition rather than binary raster operations.

Using simple compositing operators provided by most hardware, Render can draw anti-aliased text and geometric objects as well as perform translucent image overlays and other image operations not possible with the core X rendering system.

XFree86 4.6.0 provides a partial implementation of Render sufficient for drawing anti-aliased text and image composition. Still to be implemented are geometric primitives and affine transformation of images.

Unlike the core protocol, Render provides no font support for applications, rather it allows applications to upload glyphs for display on the screen. This allows the client greater control over text rendering and complete access to the available font information while still providing hardware acceleration. The Xft library provides font access for Render applications.

6.11.1 The Xft Library

On the client side, the Xft library provides access to fonts for applications using the FreeType library, version 2. FreeType currently supports Type1 and TrueType font files, a future release is expected to support BDF and PCF files as well, so Render applications will have access to the complete range of fonts available to core applications. One important thing to note is that Xft uses the vertical size of the monitor to compute accurate pixel sizes for provided point sizes; if your monitor doesn't provide accurate information via DDC, you may want to add that information to `XF86Config`.

To allow a graceful transition for applications moving from core text rendering to the Render extension, Xft can use either core fonts or FreeType and the Render extension for text. By default, Xft is configured to support both core fonts and FreeType fonts using the supplied version of FreeType 2. See the section on FreeType support in Xft for instructions on configuring XFree86 to use an existing FreeType installation.

The Xft library uses a configuration file, `XftConfig`, which contains information about which directories contain font files and also provides a sophisticated font aliasing mechanism. Documentation for that file is included in the Xft(3) man page.

6.11.2 FreeType support in Xft

XFree86 4.6.0 includes sources for FreeType version 2.1.8, and, by default, it is built and installed automatically.

6.11.3 Application Support For Anti-Aliased Text

Only three applications have been modified in this release to work with the Render extension and the Xft and FreeType libraries to provide anti-aliased text. Xterm, xditview and x11perf. Migration of other applications may occur in future releases.

By default, xterm uses core fonts through the standard core API. It has a command line option and associated resource to direct it to use Xft instead:

- `-fa` family / `.VT100.faceName:` family. Selects the font family to use.

Xditview will use Xft instead of the core API by default. X11perf includes tests to measure the performance of text rendered in three ways, anti-aliased, anti-aliased with sub-pixel sampling and regular chunky text, but through the Render extension, a path which is currently somewhat slower than core text.

6.12 Other extensions

The XFree86-Misc extension has not been fully ported to the new server architecture yet. This might be completed in a future release.

The XFree86-VidModeExtension extension has been updated, and mostly ported to the new server architecture. The area of mode validation needs further work, and the extension should be used with care. This extension has support for changing the gamma setting at run-time, for modes where this is possible. The `xgamma` utility makes use of this feature. Compatibility with the 3.3.x version of the extension is provided. The missing parts of this extension and some new features should be completed in a future release.

6.13 xedit

Xedit has several new features, including:

- An embedded lisp interpreter that allows easier extension of the editor.
- Several new syntax highlight modes, and indentation rules for C and Lisp.
- Flexible search/replace interface that allows regex matches.
- Please refer to `xedit(1)` for more details.

6.14 Font support

Details about the font support in XFree86 4.x can be found in the README.fonts document.

6.15 TrueType support

The XFree86 4.x server comes with a TrueType backend, known as the "FreeType" backend, based on the FreeType library. The functionality of the former X-TrueType was merged into the FreeType backend by the After X-TT Project for the XFree86 4.4.0 release, and the old X-TrueType backend has been dropped as of XFree86 4.5.0.

6.16 CID font support

Support for CID-keyed fonts is included in XFree86 4.x. The CID-keyed font format was designed by Adobe Systems `<URL:http://www.adobe.com>` for fonts with large character sets. The CID-keyed font support in XFree86 was donated by SGI `<URL:http://www.sgi.com>`. See the LICENSE document for a copy of the CID Font Code Public License.

6.17 Internationalisation of the scalable font backends

XFree86 4.x has a ''fontenc'' layer to allow the scalable font backends to use a common method of font re-encoding. This re-encoding makes it possible to use fonts in encodings other than their

their native encoding. This layer is used by the FreeType, Type1 and Speedo backends.

6.18 Large font optimisation

The glyph metrics array, which all the X clients using a particular font have access to, is placed in shared memory, so as to reduce redundant memory consumption. For non-local clients, the glyph metrics array is transmitted in a compressed format.

6.19 Unicode/ISO 10646 support

What is included in 4.x:

- All ''-misc-fixed-*'' BDF fonts are now available in the ISO10646-1 encoding and cover at least the 614 characters found in ISO 8859-{1-5,7-10,14,15}, CP1252, and MES-1. The non-bold fonts also cover all Windows Glyph List 4 (WGL4) characters, including those found in all 8-bit MS-DOS/Windows code pages. The 8-bit variants of the ''-misc-fixed-*'' BDF fonts (ISO8859-1, ISO8859-2, KOI8-R, etc.) have all been automatically generated from the new ISO10646-1 master fonts.
- Some ''-misc-fixed-*'' BDF ISO10646-1 fonts now cover a comprehensive Unicode repertoire of over 3000 characters including all Latin, Greek, Cyrillic, Armenian, Gregorian, Hebrew, IPA, and APL characters, plus numerous scientific, typographic, technical, and backwards-compatibility symbols. Some of these fonts also cover Arabic, Ethiopian, Thai, Han/Kanji, Hangul, full ISO 8859, and more. For the 6x13 font there is now a 12x13ja Kanji extension and for the 9x18 font there is a 18x18ja Kanji/Han/Hangul extension, which covers all ISO-2022-JP-2 (RFC 1554) characters. The 9x18 font can also be used to implement simple combining characters by accent overstriking. For more information, read Markus Kuhn's UTF-8 and Unicode FAQ `<URL:http://www.cl.cam.ac.uk/~mgk25/unicode.html>`.
- Mark Leisher's ClearlyU proportional font (similar to Computer Modern).
- ISO 10646/Unicode UTF-8 Level 1 support added to xterm (enabled with the `-u8` option).
- The FreeType font backend supports Unicode-encoded fonts.

6.20 Xlib Compose file support and extensions

A more flexible Compose file processing system was added to Xlib in XFree86 4.4.0. The compose file is searched for in the following order:

1. If the environment variable **$XCOMPOSEFILE** is set, its value is used as the name of the Compose file.
2. If the user's home directory has a file named "`.XCompose`", it is used as the Compose file.
3. The old method is used, and the compose file is "<*xlocaledir*>`/`<*localename*>`/Compose`".

Compose files can now use an "include" instruction. This allows local modifications to be made to existing compose files without including all of the content directly. For example, the system's iso8859-1 compose file can be included with a line like this:

```
include "/usr/X11R6/lib/X11/locale/iso8859-1/Compose"
```

There are two substitutions that can be made in the file name of the include instruction. **%H** expands to the user's home directory (the **$HOME** environment variable), and **%L** expands to the name of the locale specific Compose file (i.e., "<*xlocaledir*>`/`<*localename*>`/Compose`").

For example, you can include in your compose file the default Compose file by using:

```
include "%L"
```

and then rewrite only the few rules that you need to change. New compose rules can be added, and previous ones replaced.

Finally, it is no longer necessary to specify in the right part of a rule a locale encoded string in addition to the keysym name. If the string is omitted, Xlib figures it out from the keysym according to the current locale. I.e., if a rule looks like:

```
<dead_grave> <A> : "\300" Agrave
```

the result of the composition is always the letter with the "\300" code. But if the rule is:

```
<dead_grave> <A> : Agrave
```

the result depends on how Agrave is mapped in the current locale.

6.21 Luxi fonts from Bigelow and Holmes

XFree86 includes the ''Luxi'' family of Type 1 fonts and TrueType fonts. This family consists of the fonts ''Luxi Serif'', ''Luxi Sans'' and ''Luxi Mono'' in Roman, oblique, bold and bold oblique variants. The TrueType version have glyphs covering the basic ASCII Unicode range, the Latin 1 range, as well as the *Extended Latin* range and some additional punctuation characters. In particular, these fonts include all the glyphs needed for ISO 8859 parts 1, 2, 3, 4, 9, 13 and 15, as well as all the glyphs in the Adobe Standard encoding and the Windows 3.1 character set.

The glyph coverage of the Type 1 versions is somewhat reduced, and only covers ISO 8859 parts 1, 2 and 15 as well as the Adobe Standard encoding.

The Luxi fonts are original designs by Kris Holmes and Charles Bigelow from Bigelow and Holmes Inc., who developed the Luxi typeface designs in Ikarus digital format. URW++ Design and Development GmbH converted the Ikarus format fonts to TrueType and Type 1 font programs and implemented the grid-fitting "hints" and kerning tables in the Luxi fonts.

The license terms for the Luxi fonts are included in the file 'COPYRIGHT.BH', as well as in the License document. For further information, please contact <design@bigelowandholmes.com> or <info@urwpp.de>, or consult the URW++ web site <URL:http://www.urwpp.de>.

7. Credits

This section lists the credits for the XFree86 4.6.0 release. For a more detailed breakdown, refer to the CHANGELOG file in the XFree86 source tree, the cvs-commit archives <URL:http://www.mail-archive.com/cvs-commit@xfree86.org/maillist.html>, or the 'cvs log' information for individual source files.

New Features, Enhancements and Updates:

Security Updates:
Marc La France.

ASPEED Technologies driver:
Y. C. Chen, ASPEED Technologies.

Xabre Graphics driver:
Jong Lin, XGI.

X server extension rework:
Marc La France.

Loader Enhancements:
David H. Dawes, X-Oz Technologies.

Intel i945G/i945GM support and other driver enhancements:
Alan Hourihane, Tungsten Graphics.

NVIDIA driver updates and new hardware support:
Mark Vojkovich

Improved SunOS/SPARC and SBUS support:
Marc La France.

Xterm enhancements and updates:
Thomas Dickey.

Xaw enhancements:
Alexander Pohoyda.

Integration:

General Integration of Submissions:
Marc La France, David H. Dawes.

Release Engineering:
David H. Dawes.

Patches and other submissions (in alphabetical order):
ASPEED Technologies, Andrew Aitchison, James Ascroft-Leigh, Étienne Bersac, Peter Breitenlohner, Terry Chang, Y. C. Chen, Jeff Chua, James Cloos, Alan Coopersmith, Miguel González Cuadrado, David Dawes, Thomas Dickey, Stefan Dirsch, Bernd Ernesti, Jordan Frank, Will L G, Frank Giessler, Thorsten Glaser, Damian Janusz Gruszka, Lukas Hejtmanek, Evil Mr Henry, Jens Herden, Alan Hourihane, Nicolas Joly, Bang Jun-Young, Alexander Kabaev, Satoshi Kimura, Milos Komarcevic, Marc La France, Dejan Lesjak, Khong Jye Liew, Jong Lin, Michael Lorenz, Michael Macallan, Michal Maruska, Luke Mewburn, Timothy Musson, Newsh, Takaaki Nomura, Ivan Pascal, Bob Peterson, Pierre, Aaron Plattner, Alexander Pohoyda, Jeremy C. Reed, Conrad Schuler, Bruno Schwander, Olaf Seibert, Aaron Solochek, Helmar Spangenberg, Ken Stailey, Tobias Stoeckmann, Tungsten Graphics, James Richard Tyrer, Staffan Ulfberg, Denis Vlasenko, Mark Vojkovich, Tom Williams, Dave Williss, X-Oz Technologies, XGI, Christos Zoulas.

Webmaster:
Georgina O. Economou

Hosting:
Internet Systems Consortium, Inc <URL:http://www.isc.org/>.

8. Attributions/Acknowledgements

This product includes software developed by The XFree86 Project, Inc (http://www.xfree86.org/) and its contributors.

This product includes software that is based in part of the work of the FreeType Team (http://www.freetype.org).

This product includes software developed by the University of California, Berkeley and its contributors.

This product includes software developed by Christopher G. Demetriou.

This product includes software developed by the NetBSD Foundation, Inc. and its contributors

This product includes software developed by X-Oz Technologies (http://www.x-oz.com/).

CONTENTS

$XFree86: xc/programs/Xserver/hw/xfree86/doc/sgml/RELNOTES.sgml,v 1.142 2006/04/29 03:35:46 dawes Exp