

# Freedesktop: Hacia un escritorio común.

Alejandro Sánchez Acosta<sup>12</sup>

<sup>1</sup> GNU.

asanchez@gnu.org,

WWW Home Page: <http://www.advogato.org/person/alejandro/>

<sup>2</sup> Grupo S21Sec Gestión S.A,

asanchez@s21sec.com,

WWW Home Page: <http://www.s21sec.com>

**Resumen** Este artículo pretende cubrir el proyecto Freedesktop e intentar inculcar tanto a usuarios como desarrolladores lo importante que es tener un escritorio estandarizado e integrado, no sólo cubriendo la necesidad del escritorio, sino cubriéndolo desde un bajo nivel hasta un alto nivel, consiguiendo una alta integración e interoperabilidad entre las distintas tecnologías posibles que se han ido recogiendo y estandarizando dentro del proyecto Freedesktop.

## 1. Introducción

Freedesktop.org, también conocido como fdo, es un proyecto de software libre iniciado en el 2000 por la iniciativa de uno de los desarrolladores de GNOME, Havoc Pennington, para la estandarización, integración e interoperabilidad de los distintos escritorios.

Entre los escritorios más concurridos que se tratarán serán KDE, GNOME y XFCE, al igual que otros gestores de ventanas, aunque cualquier otro escritorio existente también puede ser reflejado dentro del proyecto.

Para ello Freedesktop lo que quiere conseguir es una base de software común para todos los escritorios, englobando tecnologías basadas en Mozilla (XUL, XBEL, GECKO), distintos toolkits gráficos (QT, GTK+ y VCL), WINE, GNOME, KDE y X Window System.

## 2. Objetivos de Freedesktop

Freedesktop surgió principalmente como un proyecto entorno al sistema X para la estandarización, integración e interoperabilidad de los distintos escritorios de GNU/Linux con el fin de conseguir que tecnologías existentes estén más integradas, de forma que no haya una total diversidad y separación entre proyectos.

Dentro del marco de un usuario final, lo que este quiere encontrar es un escritorio integrado, que siga unas guías de usabilidad y que interopere con las

tecnologías ya existentes, dándole la sensación de un entorno totalmente integrado. Además de cara al programador es necesario unos estándares y especificaciones que se vayan desarrollando para su uso entre los diversos escritorios, con la decisión de una organización, grupo o comité que se encargue de realizar estas especificaciones e incluso llegar a implementarlas.

Entre los objetivos más importantes marcados por el proyecto están los siguientes:

- Reunir especificaciones, estándares y documentos relacionados con el escritorio para su interoperabilidad y uso centralizado.
- Promover el desarrollo de estos.
- Integrar estos estándares en iniciativas como ICCCM o LSB.
- Un foro neutral para la discusión de distintos escritorios entre desarrolladores y usuarios.
- Implementación de dichas tecnologías.
- Promover los escritorios X al igual que el sistema X Window.
- Mayor comunicación entre desarrolladores de sistemas operativos libres, de X Window y de los propios escritorios para conseguir una integración completa teniendo una pila entera sin capas desde el kernel hasta el propio escritorio.

### 3. Estandarización

Freedesktop en sí no se trata de una organización de estándares como Linux Standard Base (LSB), la IETF o en sí el consorcio X, sino que es una comunidad de colaboraciones donde tanto nuevas ideas como código puede ser albergado, dando lugar a especificaciones y estándares para el escritorio.

Por ello, para estándares oficiales se tendrá que recurrir a otros proyectos, pero sí que se recogen estándares ya recogidos por algunas tecnologías de escritorio ampliamente usadas en algunos proyectos de escritorio como KDE o GNOME.

Entre ellos los más destacados son:

- **CORBA o (Common Object Request Broker Architecture)**: es utilizado por la plataforma GNOME para la comunicación entre procesos como mecanismo IPC. Sería similar a la tecnología COM+ o a DCOP en KDE.
- **XML**: Es una forma de estructurar los datos usado tanto en KDE como GNOME.
- **Document Object Model (DOM)**: es usado para la manipulación de documentos XML en memoria.
- **X Window System**: Es el modelo cliente/servidor para la representación gráfica en pantalla. Recoge varios estándares como ICCCM, EWMH, protocolo de sesión (SMP) o Smlib.
- **Especificación de UNIX**: Contiene las interfaces de programación clásicas de UNIX (mecanismos IPC, semáforos, comunicación por red, etc).

Además se han llegado a recoger algunos borradores ampliamente utilizados como:

- **XDND**: Es la especificación sobre la que se basan los toolkits gráficos para tener soporte para soltar y arrastrar entre diferentes aplicaciones y ventanas.
- **Especificación del gestor de ventanas (window manager)**: Se encarga de la estandarización de nuevas características sobre la especificación ICCCM. También estas especificaciones son conocidas como EWMH (Extended Window Manager Hits).
- **Entradas de escritorio Desktop**: Para la representación de iconos o lanzadores se ha utilizado esta especificación. En esta se incluye la descripción de ficheros, uso de lanzadores y creación de menús. Actualmente se ha estandarizado tanto en GNOME como KDE.
- **Xembed**: Se ha utilizado para incrustar/empotrar controles dentro de aplicaciones y se trata de una extensión sobre el sistema X Window.
- **X Clipboard**: Trata del funcionamiento del portapapeles con el fin de poder cortar/pegar entre diferentes aplicaciones. Es aplicable a toolkits gráficos como QT y GTK+.
- **UTF-8**: Especificación UNICODE para la codificación de caracteres. Se ha acabado implantando en el escritorio como codificación por defecto.
- **XBEL**: Es una especificación para el manejo/intercambio de bookmarks/enlaces entre diferentes aplicaciones, concretamente más los navegadores web. Se utiliza principalmente en los proyectos Mozilla (Mozilla/Firefox).

Luego existen otras especificaciones usadas por algún escritorio como:

- **Especificación del menú**: Trata sobre como crear menús a partir de entradas Desktop. Adoptada tanto en KDE como GNOME.
- **Desktop base directory**: Especifica el directorio base donde se van a guardar todos los ficheros desktops. Recientemente adoptada junto con la especificación de menús y .desktop.
- **Shared MIME DB**: Describe como identificar tipos de ficheros a través de tipos MIME y es usada por los selectores de ficheros. Así se puede saber de forma unívoca que aplicaciones tienen que tratar determinados tipos de ficheros, de forma que se asocie varias aplicaciones por tipo MIME, tanto de KDE como GNOME.
- **XSETTINGS**: Proporciona un mecanismo para que las aplicaciones escritas en diferentes toolkits gráficos puedan compartir configuraciones de usuario.

Al haber un gran amplio de especificaciones no se ha tratado en detalle cada una de ellos, el objetivo es unicamente poder conocer cada uno de los estándares, especificaciones y borradores.

#### 4. Extensiones de las X

Son muchas las extensiones y especificaciones entorno a las X, aquí podríamos hablar incluso de gran parte de la arquitectura del Sistema X Window como

son las xlib, la reescritura que se pretende a hacer para conseguir unas xlib más ligeras y compatibles, extensiones como XDAMAGE, XCOMPOSITE y XFIXES.

Además hay especificaciones para el manejo de cortar/pegar con el clipboard, la extensión Xembed para la incrustación de controles en nuestras aplicaciones, XdnD para soltar arrastrar elementos entre aplicaciones y el manejo de sesiones.

Entre las más destacadas e importantes están:

**XCOMPOSITE:** Para un usuario final esta extensión será la más beneficiosa ya que es el que se encarga de la visualización de contenidos en el escritorio permitiendo transparencias y sombras.

Basicamente sus funcionalidades son las siguientes:

- 1. Almacenamiento de objetos por jerarquía: Para el renderizado de objetos/widgets en pantalla es necesario que se guarde una jerarquía en memoria secundaria para su manipulación. Cada vez que se realiza un cambio es guardado en memoria secundaria el elemento en pantalla con la modificación oportuna antes de proceder a su visualización.
- 2. Actualización de sombras de forma automática: Una vez que la jerarquía es guardada en memoria secundaria el servidor X automaticamente proporciona un mecanismo para mostrar estos contenidos en pantalla.

**XDAMAGE:** Permite la notificación de cambios sobre regiones/zonas de cualquier objeto representable en las X. Es muy útil para sesiones VNC o sesiones cliente/servidor ya que permite reducir tiempos de latencia en el cliente y ancho de banda.

Se define daño o damage como aquella región que ha sido modificada. Se entiende como modificado aquellos contenidos de ventanas o pixmapes que han tenido una variación. La extensión DAMAGE notifica de estas variaciones mediante un conjunto de eventos al servidor.

**XFIXES:** Se trata de una extensión que soluciona la interacción de aplicaciones con el entorno de ventanas, de forma que sea portable sin la necesidad de contribuir código dentro del propio core de las X.

Algunos de los cambios que se han hecho son relacionados con Xembed, debido a que los nuevos módulos requerían cambios dentro del sistema de ventanas. En el caso de XEmbed define la posibilidad de empotrar una aplicación dentro de otra pero esto no es reflejado dentro del gestor de ventanas lo que causa que a la hora de definir como objetivo la ventana a la que queremos empotrar y consecuentemente mata esa ventana. Xfixes soluciona este problema definiendo como target la ventana destino.

En el caso de la manipulación y monitorización de imágenes/cursores para el ratón/puntero se suele realizar mediante el protocolo origen de las X o por añadidos de VNC. El problema está en que no hay manera de saber que imagen del cursor se está actualmente visualizando. Mediante xfixes se proporciona un servicio de descubrimiento de los cambios de imágenes que está utilizando un cursor.

Esta extensión también es necesaria para el manejo de regiones con Damage ya que no está contemplada la notificación de cambios en regiones a nivel del

propio protocolo X, así que es necesario que xfixes se encargue de ello. Esto es muy útil luego a la hora de utilizar sombras y transparencias.

Otras posible característica es la posibilidad de asignar nombres a los cursores.

**XEVENT** interception: También conocido como XEvIE es utilizado para el control de eventos tanto de teclado como ratón y viene por una necesidad en ATK, el proyecto de accesibilidad en GNOME, de tener control total sobre los periféricos que usa el usuario.

**XDND**: La especificación de DND es un requerimiento importante para la comunicación entre aplicaciones mediante la posibilidad de poder arrastrar/soltar elementos de una aplicación a otra. El motivo de esta extensión de las X es que ha habido varias implementaciones y lo que se ha buscado es tener un estándar común para la interoperabilidad entre distintos toolkits gráficos.

**XEMBED**: Esta extensión nos permite poder incrustar aplicaciones dentro de otras.

Estas jugarán un papel muy importante sobre nuevos servidores como xizze y xserver, que nos permitirá características como animación de ventanas, tipo expocity y expose y uso de transparencias y sombras en las X.

## 5. Proyectos de software en Freedesktop

Según ha ido evolucionando el proyecto se han ido albergando más proyectos dentro de freedesktop consiguiendo formar un repositorio bastante amplio de software. Entre los proyectos más destacados están los siguientes:

- **Cairo** es una librería de renderizado de gráficos que se apoya en Xrender para la generación de imágenes. Se acabará adoptando en lugar de gdk.
- **CFG modular configuration management**, handling arbitrary config files/databases.
- **ChIX** permite el manejo de la codificación china en el servidor X.
- **D-BUS** es un mecanismo IPC para la comunicación de aplicaciones tanto local como distribuida parecido a DCOP y CORBA.
- **Desktop-file-utils** contiene utilidades para el manejo de entradas desktop y menu.
- **DRI** es un framework para la comunicación directa con la tarjeta gráfica.
- **Enchant** es la integración de varios correctores ortográficos (myspell, ispell, ..)
- **GTK-Qt-Engine** es un proyecto para la integración de temas entre gtk y qt.
- **Fontconfig** es una librería para la configuración, detección y automatización de tipos de fuentes.
- **FriBidi** es una librería que permite usar unicode de forma bidireccional para lenguas como el árabe.
- **Gstreamer** es un framework multimedia usado en el proyecto GNOME.
- **HAL** es una implementación de la capa de abstracción hardware, se aprovecha de sysfs para obtener información del hw en el sistema.

- **Icon-slicer** es una herramienta para la generación de iconos.
- **Icon-theme** es un estándar para la especificación de los temas de iconos.
- **Jhbuild** es un programa para la compilación de determinadas aplicaciones del CVS, entre ellas del proyecto GNOME y fdo.
- **Libburn** es una librería que pretende estandarizarse en los escritorios para la grabación de unidades ópticas en el escritorio.
- **Libxklavier** es una librería para el manejo de eventos de teclado.
- **Media Application Servertm (MAS<sup>TM</sup>)** es un servidor multimedia sustituto de arts y esound.
- **Pkgconfig** sirve para la compilación de programas, especifica de forma automática las librerías y los flags a utilizar.
- **Pyxdg** es una librería de python para el acceso a los estándares de xdg.
- **Glitz** es una librería de gráficos 2D opengl y un backend para la salida GL en Cairo.
- **SCIM** es otro proyecto para la entrada de diferentes codificaciones de caracteres de forma unificada (XIM, IIMF, kinput, XCIN o Chinput.).
- **Shared-mime-info** contiene los tipos MIME para su uso dentro del escritorio.
- **UTF-8** es un proyecto para el uso de la codificación de caracteres UTF-8.
- **Unicode-translation** es para la traducción de caracteres de nombres UNICODE a otros lenguajes.
- **XCB/XCL** es un intento de reescritura de las librerías X.
- **XFullscreen** es un módulo que permite la visualización en pantalla completa de las aplicaciones para su integración en el gestor de ventanas.
- **X Libraries** contiene las librerías de las X.
- **XRestop** monitoriza los recursos de las X utilizando la extensión XRes.
- **X Server** también conocido como tinyx o kdrive, es el servidor X que realizó Keith Packard de forma modular con soporte de transparencias y sombras.
- **XSettings** implementación de Xsettings.
- **XOrg** fundación Xorg para la implementación del sistema X Window.
- **XPrint** manejo de impresión en las X.
- **XApps** contiene las aplicaciones principalmente utilizadas dentro del proyecto X de forma modular y usando las autotools en vez del clásico imake.

## 6. Implementaciones del servidor X

Uno de los desarrolladores quizás más conocidos del proyecto es Keith Packard, uno de los desarrolladores de X Window System, que decidió dejar de colaborar por diversos motivos y mantener su rama de tiny server/xserver/kdrive.

Este servidor incorpora nuevas características como el uso de transparencias y sombras. Actualmente se está trabajando para su soporte en diferentes tarjetas gráficas, dando la posibilidad incluso de aceleración en tarjetas como Ati Mach64. En el resto de casos se tendrá que seguir usando el driver de vesa y el servidor Xvesa que proporciona kdrive.

Aparte del servidor, para incluir estas mejoras se ha necesitado el uso de nuevas extensiones como C/D/M (composite, damage y fixes) para el uso de transparencias y sombras. Además se trata de un servidor muy pequeño, por lo que cabe la posibilidad de empotrarse en dispositivos empotrados.

Aún así no sólo ha habido esta implementaciones de las X dentro de fdo, sino que se encuentran otros dos servidores. Uno de ellos es XIZZLE que es un servidor X en el que se han ido incorporando las características de C/D/F pero con toda la funcionalidad del sistema X Window.

Luego con el cambio de licencia de las X, se ha hecho un fork del proyecto que se encuentra bajo el módulo x.org.

Entre las características que se irán incluyendo son:

- Poder hacer zoom a una ventana.
- Uso de expose.
- Movimientos y cambio de tamaño de forma translúcida.
- Transparencias de ventanas con el uso del canal alfa.
- Interacción con el gestor de ventanas.
- Gestor de composite o xcompmanager.
- Integración 3D.

## 7. Nuevos proyectos

Según ha ido creciendo el proyecto freedesktop se han ido incorporando nuevos proyectos. Entre los proyectos más recientes se destacan:

- **Avahi:** Librería con soporte zeroconf/mdns para descubrimiento de servicios. Actualmente tanto en KDE como GNOME las implementaciones están basadas en mdns.
- **Clipart:** Librería con una amplia cantidad de cliparts. Principalmente ahora se utiliza dentro del proyecto OpenOffice.
- **Gnome-system-tools backends:** Recoge los backends utilizados en gnome-system-tools para la administración de distintas distribuciones de forma genérica.
- **Galago:** Sistema de notificación recogida en GNOME. Actualmente no se ha llegado a estandarizar su uso debido a discrepancias por no usar directamente la implementación previamente existente en KDE conocida como knotify.
- **Poppler:** Se considera un fork de xpdf para el uso como librería en visores de PDF. Actualmente se utiliza tanto en KPDF como en Evince.
- **Xgl:** Sistema X Window utilizando tecnología OpenGL.

## 8. Integración entre sistemas operativos

También dentro del proyecto forma parte todo aquel software que sirva para la integración entre diversos sistemas operativos. Entre ellos cabe destacar:

- **WINE:** Emulación de aplicaciones de Windows en sistemas abiertos.

- **CUPS:** Sistema de impresión genérico para diferentes sistemas operativos.
- **Migración de aplicaciones a Windows:** Algunos de los proyectos más destacados son Firefox y Thunderbird. Además se está intentando realizar lo mismo con proyectos como Evolution, intentando llevar la tecnología de escritorios como GNOME/KDE a entornos propietarios.
- **Máquinas virtuales:** Extensión de uso de sistemas operativos integrando tecnologías basadas en Xen, User-mode-linux, Qemu o Bochs.

## 9. Uso de tecnologías en GNOME/KDE.

Posteriormente explicado cada uno de los proyectos, se observa una repercusión importante del proyecto dentro del escritorio.

Desde el punto de vista de KDE, se pueden comentar las siguientes mejoras:

- Se ha adoptado el uso de DBUS como tecnología de comunicación IPC dentro de Qt4.
- Se ha empezado a utilizar gstreamer como framework multimedia dentro de aplicaciones como Amarok/Juk.
- Uso de .desktop para el uso de menús y aplicaciones.
- Plopper como librería para render de PDF's en KPDF.
- Uso de estándares basados en DOM, XML, XBEL y HTML en KHTML/Konqueror.
- Mayor integración entre aplicaciones y estudio de usabilidad.
- Creación de APPEAL para impulsar el escritorio de KDE (creación de Tora), conseguir más usabilidad y que el escritorio sea más eye-candy.
- Hotplugging mediante sistema HAL.

En el estado del escritorio de GNOME se observa:

- Se empieza a adoptar DBUS para la comunicación entre aplicaciones.
- Mejora en menús y especificación de aplicaciones mediante .desktop generalizado.
- Uso de estándares basados en DOM, XML, UTF-8, entre otros.
- Creación un visor PDF basado en Poppler.
- Intentar un escritorio usable, intentando conseguir también una mayor integración y comunicación entre aplicaciones.
- Uso de Galago para la notificación de eventos.
- Comienzo del reconocimiento de dispositivos mediante proyecto UTOPIA (HAL y DBUS).

## 10. Futuro.

Dentro del futuro de los escritorios irán surgiendo nuevas tecnologías. En mi opinión parte del trabajo irá destinado a:

- Búsquedas de ficheros por Metadatos: Beagle y Tora.

- Mayor integración entre aplicaciones.
- Mejora en eye-candy: Xgl, X Window System con extensiones, Expose, transparencias, sombras, menus, tooltips mejorados, etc.
- Misma apariencia entre toolkits.
- Mayor usabilidad.

## 11. ¿Cómo involucrarse?

Como otros proyectos de software libre la mejor forma de colaborar es conociendo el proyecto y poco a poco irse involucrando.

En este tema, la mejor prueba sería ir investigando cada uno de los módulos de CVS en la que tenemos una sencilla interfaz web con viewcvs o bonsai. Además podemos bajar cada uno de los módulos con:

```
cvs -d:pserver:anoncvs@pdx.freedesktop.org:/cvs login
cvs -d:pserver:anoncvs@pdx.freedesktop.org:/cvs co modulo
```

Aparte por cada proyecto disponemos de una lista de correo. Como lista general está la de xdg que es la de freedesktop:

```
http://freedesktop.org/mailman/listinfo
```

Si queremos consultar cualquier pregunta a los desarrolladores, también se puede uno conectar a la red de freenode (irc.freenode.org) al canal freedesktop.

Para la consulta y reporte de bugs podemos acudir a:

```
http://bugzilla.freedesktop.org.
```

## Referencias

1. Proyecto Freedesktop: <http://www.freedesktop.org>
2. Proyecto GNOME: <http://www.gnome.org>
3. Proyecto KDE: <http://www.kde.org>
4. Desarrollo en GNOME: <http://developer.gnome.org>
5. Desarrollo en KDE: <http://developer.kde.org>
6. Blog de desarrolladores de fdo: <http://planet.freedesktop.org>
7. Blog de desarrolladores de GNOME: <http://planet.gnome.org>
8. Blog de desarrolladores de KDE: <http://www.kdedevelopers.org>