

## A Full Working Environment Based on a Thin Client

◆ J. A. Marín y L. Meléndez

### Resumen

De entre las distintas soluciones posibles para gestionar aulas informatizadas con cientos de PC, presentamos una alternativa basada en un cliente ligero (dada su facilidad de despliegue y posibilidades de conexión a una granja de servicios de terminal Windows), incorporándole los elementos necesarios para hacer posible su uso simultáneo como entorno autónomo de trabajo basado en GNU/Linux: servicio de nombres y de autenticación basado en red, proceso de login, aplicaciones accesible por red, etc. En resumen: ejecución local de Linux sin disco duro y acceso remoto a servidores Unix y/o Windows.

**Palabras clave:** Thinstation, aulas de ordenadores, clientes ligeros, GNU/Linux, Windows

### Summary

Among several possible solutions about computer-rooms management problem with hundreds of PCs, we propose an alternative based on Thin Clients (given its ease of deployment as well as its ability to connect to Windows terminal services farms), adding necessary elements to make possible simultaneous use as a GNU/Linux autonomous working environment: network-based name and authentication services, login process, network-mounted applications, etc. In brief: local execution of Linux without hard drive and remote access to Unix and/or Windows servers.

**Keywords:** Thinstation, computer rooms, thin clients, GNU/Linux, Windows

## 1. Introducción

El mantenimiento de equipos informáticos en aulas (y en general de grandes parques de ordenadores) ha sido siempre un quebradero de cabeza para los administradores de sistemas, por varias razones conocidas por todos. En el caso de las aulas universitarias, el asunto se complica aún más debido a factores específicos:

- Normalmente hay que proporcionar los entornos y aplicaciones que demanden los docentes.
- Suele darse una gran heterogeneidad en los equipos (distintos momentos de adquisición, dominios de competencias, etc.).
- El entorno es muy dinámico. Las necesidades docentes van evolucionando constantemente.
- Multitud de usos posibles: docencia de disciplinas muy diversas, libre disposición, cursos, jornadas, etcétera.
- Geográficamente pueden estar muy dispersas en los diferentes campus y edificios.

La solución técnica que se adopte en un entorno de este tipo estará sometida asimismo a otros condicionantes tan importantes como el coste económico y la disponibilidad de recursos humanos. Además existen otros menos tangibles, como pueden ser las tendencias a favor de la implantación y/o fomento del *software* libre (a veces incluso hasta extremos poco realistas por intereses políticos o de imagen).

No nos ocuparemos en este artículo de otros aspectos más 'de gestión' que también son importantes: las bases de datos de equipos, el inventariado de *hardware*, etc.



El mantenimiento de equipos informáticos en aulas ha sido siempre un quebradero de cabeza para los administradores de sistemas



La solución técnica que se adopte en este entorno estará sometida a otros condicionantes: el coste económico y la disponibilidad de recursos humanos



◆  
Siempre hemos desechado cualquier sistema basado en la instalación en el disco de los equipos. El disco local es una fuente de fallos

◆  
Pensamos que no es viable imponer una plataforma de trabajo o determinados programas si no se proporciona a los usuarios la posibilidad de utilizarlos de forma alternativa con los que están acostumbrados

## 2. Alternativas

Mencionaremos brevemente las principales alternativas para implantar un sistema de aulas:

1. Equipos autónomos gestionados individualmente. Totalmente descartado por inabordable.
2. Equipos autónomos con replicación de imágenes. Un modelo bastante popular.
3. Equipos autónomos con *software* a través de red. El sistema se ejecuta localmente, pero las aplicaciones se cargan de la red. No es aplicable a todos los sistemas operativos y/o aplicaciones.
4. Thin clients.
5. Entornos mixtos.

No hay una solución universal. En cada institución hay que valorar las necesidades, los requisitos, condicionantes, etc.

El modelo que describimos en este artículo es el que se utiliza actualmente en prácticamente todas las aulas informáticas de la Universidad de Córdoba, así como algunos colectivos del PAS.

## 3. Antecedentes

Siempre hemos desechado cualquier sistema basado en la instalación en el disco de los equipos, por muy automatizada que esté. El disco local es una fuente de fallos, escapa a nuestro control directo y no podemos saber en cada momento qué ha pasado con determinado equipo y si funciona o no igual que el resto.

En nuestro caso la búsqueda de una evolución del modelo anterior (basado en Metaframe, RPL y cliente ICA de 16 bits) vino motivada principalmente por las limitaciones del cliente ICA para MSDOS y Windows 3.11. Era necesario disponer de un S.O. que permitiera versiones más modernas del mismo. Por otra parte, queríamos utilizar las CPU locales y se nos había pedido que se pudiera utilizar Linux.

La tendencia a impulsar el uso de *software* libre en general y GNU/Linux en particular, aparte de ser bastante lógica, viene impulsada desde todas las instancias políticas y de decisión. Nuestra Universidad no es una excepción.

En nuestro caso, se solicitó al Área de Sistemas que las aulas pudieran contar con un entorno de trabajo GNU/Linux. Con la situación de recursos humanos en ese momento, no lo consideramos posible. Si bien durante muchos años se ha utilizado extensivamente aquí el acceso remoto a servidores SunOS y Solaris, replicar el modelo con Linux no parecía lo más adecuado (coste de servidores de potencia equivalente, 'pesadez' de los entornos gráficos actuales de Linux, etc.). Sin embargo, las primeras experiencias con Thinstation [1] nos hicieron ver un camino que nos llevó en poco tiempo (relativamente) no sólo a cumplir con el objetivo propuesto, sino a un modelo que ha tenido gran aceptación entre los usuarios y que para nosotros supone una gestión bastante sencilla.

En cuanto a la tendencia por el *software* libre, pensamos que no es viable imponer una plataforma de trabajo o unos determinados programas si no se proporciona a los docentes y usuarios la posibilidad de utilizarlos de forma alternativa a los que están acostumbrados. En nuestro caso, si bien se podía prever que nuestro entorno era ideal para conseguir esto, los resultados han sido más positivos de lo esperado, como explicaremos en el apartado de conclusiones.

## 4. Punto de partida

Queríamos llegar a un sistema de fácil gestión y en el que los usuarios pudieran trabajar en Linux o Windows, a su elección. La instalación local de Windows prácticamente estaba descartada por sus múltiples complicaciones y la experiencia acumulada de años de administrar una granja de servidores Citrix, con resultados más bien positivos.

El acceso por ICA es sencillo, basta con la aplicación cliente, que existe para varios sistemas operativos, entre ellos GNU/Linux. El arranque y la ejecución de éste es lo que hay que resolver. Un cliente ligero basado en Linux es la opción más sencilla de desplegar, pero carece de la posibilidad de ejecutar cualquier tipo de aplicación local. Un sistema de replicación de imágenes con sistemas Linux lo consideramos inviable por distintos motivos. Pensamos que lo ideal sería algo intermedio: Si con un cliente ligero ya tenemos un mini-Linux (arrancado de la red, sin necesidad de disco local) funcionando en el equipo, ¿sería factible añadirle lo necesario para conseguir un Linux 'de verdad', con todas las aplicaciones necesarias accesibles por red?

De entre las diversas opciones de clientes ligeros basados en Linux, optamos por Thinstation. Nos gustó su filosofía, la sencillez y potencia de su sistema, tanto de generación de imágenes como de configuración de arranque de clientes. Todo su funcionamiento se basa en *scripts* (muy parametrizables mediante variables), fáciles de entender y de modificar si es necesario.



Un cliente ligero basado en Linux es la opción más sencilla de desplegar

## 5. Thinstation

Es un proyecto de *software* libre que nació en mayo de 2003 como evolución de otro proyecto anterior, Netstation. Sus componentes básicos son:

- Sistema de generación de imágenes. Normalmente basta editar el fichero 'build.conf' y ejecutar el comando 'build' para obtener imágenes para distintos tipos de arranque: syslinux, pxe, isolinux, etherboot, loadlin.
- Imagen de arranque. Contiene el kernel de Linux y una imagen initrd. El tamaño es crítico, tanto por la rapidez de arranque como porque debe entrar en memoria RAM. El nuestro ocupa actualmente unos 9Mb.
- Servidor de arranque. Basta con un DHCP+TFTP, aunque Thinstation puede arrancar también de disco duro, *pen drives*, etc.
- Paquetes. Los elementos opcionales que se pueden incluir en la imagen: conjuntos de *drivers*, clientes para distintos accesos remotos, etc.
- Entorno de desarrollo. Un entorno *chroot* semejante al de ejecución posterior, donde podemos compilar o generar nuevos paquetes o aplicaciones (en nuestro caso ya no fue necesario desde el momento en que instalamos en el entorno de trabajo 'real' todo lo necesario: compiladores, librerías, etc.).
- Ficheros de configuración. Sobre la base de variables y grupos de ficheros, se dispone de gran flexibilidad de configuración de equipos por aulas, por tipo de *hardware*, etc.

El sistema de paquetes es bastante limitado, y se aplica a los propios componentes de Thinstation (no a todas las aplicaciones que luego hemos instalado y se cargan por NFS y por tanto no van en la imagen).

El sistema que se ejecuta en el PC es un Linux que reside totalmente en memoria. El sistema de ficheros es del tipo *squashfs*, comprimido y de sólo lectura, con algunos ficheros enlazados a ficheros



Thinstation es un proyecto de *software* libre que nació como evolución de otro proyecto anterior, Netstation



en /tmp, que no es de sólo lectura. Dado el tamaño de la imagen y la memoria de los PC actuales, no hay ningún problema en añadir lo que haga falta a la imagen (siempre que mantengamos las aplicaciones pesadas en el servidor NFS)[2].

## 6. Modificaciones necesarias

Un cliente ligero habitualmente no contempla el uso de distintos usuarios. Se supone que éstos tienen relevancia en el sistema remoto al que se van a conectar. Si se va a poder trabajar de forma local, con acceso al directorio de trabajo del usuario, es necesario usar NSS y PAM, en nuestro caso usando LDAP.

◆  
Un cliente ligero habitualmente no contempla el uso de distintos usuarios

Igualmente necesario es presentar una pantalla de login (XDM o equivalente) con soporte de PAM que posteriormente arranque al menos un gestor de ventanas.

Como prerequisite para la cantidad de *software* que iba a ser necesario (al menos OpenLDAP para pam\_ldap, el entorno de compilación, etc.), era necesario disponer desde el principio de soporte NFS y tener todo en un servidor en vez de en la imagen que se carga en memoria; pero eso no supuso más problema que activar la opción de cliente NFS en la imagen y configurar los montajes.

Como no disponemos de espacio para entrar en muchos detalles, resumimos los cambios necesarios para conseguir el entorno que queremos:

- pam\_ldap
- XDM
- Window Manager (en nuestro caso IceWM)
- Montaje por NFS de los directorios de trabajo del usuario y de aplicaciones
- Utilidades de Linux 'completas' (inicialmente sólo se dispone de BusyBox)
- Posibilidad de usar el disco local para Swap
- Montaje de los directorio HOME por NFS o por Samba

◆  
El siguiente paso es proporcionar acceso a los usuarios desde su casa al mismo entorno de las aulas

Aparte, por supuesto, de compilar todas las aplicaciones de docencia y configurar los menús del IceWM para usarlas. Debe quedar claro que no consideramos al resultado de todo esto una 'distribución' de Linux, aunque es perfectamente posible que se pudiera instalar en otras instituciones.

Al arrancar el equipo (en menos de 1 minuto) aparece una aplicación propia, un 'selector' donde el usuario puede pinchar en tres botones que le dirigen a la sesión elegida: Linux local, cliente ICA contra nuestra granja, acceso X remoto a los servidores Solaris. De todas formas, se puede trabajar con todas esas sesiones y conmutar entre ellas con secuencias de teclas, ya que corren en servidores X diferentes.

## 7. Futuro

El siguiente paso es proporcionar acceso a los usuarios desde su casa al mismo entorno de las aulas. Ha sido una grata sorpresa para nosotros comprobar que, con muy pocos cambios, un ordenador potente arrancado de la misma forma que un PC de aula puede admitir perfectamente el trabajo multiusuario. Este servicio está disponible en pruebas y lo estará próximamente en explotación.

También se están valorando opciones como generar un DVD live o instalable en los PC domésticos; y con la nueva versión 2.2 de ThinStation, usar *cachefs*.

## 8. Conclusiones

En general el nivel de satisfacción de docentes, alumnos y administradores de sistemas es bastante alto. Cuando llega una remesa de equipos nuevos, sólo necesitamos asegurarnos de que con la imagen actual funciona la red, la tarjeta de vídeo y el sonido (y en su caso introducir los *drivers* adecuados en la imagen). Una vez comprobado eso, ya podemos estar prácticamente seguros de que todo el entorno va a funcionar ahora y en el futuro en esos equipos. No necesitamos tener continuamente prototipos de los distintos equipos.

El rendimiento es excelente. Con un buen servidor de ficheros no se nota la ausencia del disco local. La carga de OpenOffice, Eclipse o Firefox se hace igual de rápido o más que si lo tuviera.

Todo el *software* instalado hasta el momento ocupa unos 5Gb. Un inconveniente que puede llegar a ser grave con el tiempo es que las aplicaciones que instalamos no están controladas por ningún sistema de paquetes. Con el tiempo, como sabe todo el que haya mantenido los antiguos sistemas Unix de esta forma, compilando todo e instalando a */usr/local*, la situación de dependencias de librerías, actualizaciones, etc., puede provocar problemas, aunque con sentido común siempre se puede tener bajo control. Usamos *snapshots* en la NAS, de forma de se pueda rectificar rápidamente ante cierto tipo de errores.

Lo más satisfactorio es que, al disponer de un Linux tan a mano, mucha gente que nunca lo había usado ha empezado a hacerlo, y a muchos les ha gustado y usan localmente bastantes programas. Muchos docentes han decidido montar las aplicaciones para su docencia en Linux. La estética del escritorio y los menús se han dispuesto de forma que la gente no note mucha diferencia respecto a Windows, y hemos observado que bastantes usuarios se abstraen del S.O. subyacente, centrándose sólo en las aplicaciones.

## Referencias

[1] [www.thinstation.org](http://www.thinstation.org)

[2] [www.uco.es/cccsistemas/doc\\_ccc/TSuco.pdf](http://www.uco.es/cccsistemas/doc_ccc/TSuco.pdf)



El rendimiento es excelente. Con un buen servidor de ficheros no se nota la ausencia del disco local



La estética del escritorio y los menús se han dispuesto de forma que la gente no note mucha diferencia respecto a Windows

Juan Antonio Marín Beltrán  
(tonin@uco.es)  
Luis Meléndez Aganzo  
(luism@uco.es)  
Servicio de Informática  
Universidad de Córdoba