

El Proyecto TuX-Mobile

Enrique Soriano Salvador

<http://gsyc.escet.urjc.es/~esoriano>

Copyright (C) 2002 Enrique Soriano Salvador. Permitida la redistribución ilimitada de copias literales y la traducción del texto a otros idiomas, siempre y cuando se mantenga esta autorización y la nota de copyright.

Historial de revisiones

Revisión 1

25 de julio de 2002

En este artículo se expondrá el Proyecto TuX-Mobile, presentado como Proyecto de Fin de Carrera de la titulación de Ingeniería Informática en el curso académico 2001-2002, en la Universidad Rey Juan Carlos de Madrid. El proyecto fue calificado con una Matrícula de Honor por el tribunal de evaluación, y propuesto para su presentación a la V edición del Congreso Hispalinux. El proyecto TuX-Mobile pretende facilitar el uso de ordenadores portátiles en distintas redes de área local realizando la configuración de las interfaces de red de forma automática, transparente e independiente del entorno GNU/Linux.

Tabla de contenidos

1. El proyecto TuX-Mobile	1
2. Estudio de trabajos en el mismo área	3
2.1. Laptop-net	3
2.2. Laptop-netconf	5
2.3. Intuitively	5
2.4. Whereami	6
2.5. Conclusiones sobre los paquetes existentes	7
3. Objetivos: El Proyecto TuX-Mobile	7
4. Implementación	7
4.1. Identificación de entidades externas	8
4.2. Identificación de tareas principales	8
4.3. Jerarquía de módulos	8
4.4. Fichero de configuración	9
4.5. El comando spyMAC	9
5. Conclusiones	10
6. Referencias	11

1. El proyecto TuX-Mobile

Usar un ordenador portátil en diferentes redes de área local es habitual, dado que la potencia actual de estos dispositivos y su reducido tamaño y peso hace que esto sea posible. Dado que los protocolos de red comunes como TCP/IP fueron diseñados para computadoras fijas, estos ordenadores portátiles deben ser configurados cada vez que el usuario se mueve de una red de área local a otra.

El proyecto TuX-Mobile [1] propone una solución a éste problema para un sistema operativo GNU/Linux sobre redes de área local TCP/IP de tipo Ethernet, realizando dicha configuración de una forma automática y transparente para el usuario, en tres pasos básicos:

- Detectar que se ha producido un cambio en la localización del ordenador portátil: este primer paso se basa en la detección de ciertos eventos que hacen posible un cambio de localización, como la desconexión y conexión del cable de red o la inserción de una nueva tarjeta de red en el ordenador portátil. Para hacer que el sistema detecte los cambios de localización (el usuario se ha trasladado a una nueva red de área local) se han supuesto unos ciertos eventos que son habituales cuando un ordenador se mueve de un lugar a otro. Todos estos eventos se pueden controlar accediendo a la configuración del sistema operativo, aunque para ciertos modelos de tarjetas de comunicaciones el primer apartado no está disponible en la implementación de sus controladores o drivers. Los eventos son los siguientes:
 - Desconexión y conexión de un cable de red a la tarjeta de comunicaciones: Cuando se está usando una red Ethernet convencional, se debe estar conectado a través de un cable a ésta. Hemos considerado razonable suponer que cuando se desconecte y se vuelva a conectar el cable de red, sea probable que se haya producido un cambio de localización.
 - Inserción de una nueva tarjeta de comunicaciones: Igualmente se ha considerado que cuando se inserta una nueva tarjeta de comunicaciones en el ordenador es plausible que se haya producido un cambio de localización. Indistintamente de que realmente sea un cambio de ubicación, se debería hacer una reconfiguración de la interfaz de red, por lo que el sistema deberá proceder a ello.
 - Desconfiguración de una interfaz de red: También se considera que una desconfiguración de una interfaz de red es un evento que puede predecir un cambio de localización, y análogamente al apartado anterior, aunque no se haya producido este cambio, el sistema debería igualmente proceder su configuración
- Detectar dónde se encuentra el ordenador portátil en este instante: se basa en un mecanismo de espionaje del tráfico de la red a la que se ha conectado el ordenador portátil. Con este mecanismo se puede determinar en cual de las redes de área local especificadas previamente en la configuración del sistema se encuentra el ordenador portátil actualmente.
- Configurar las interfaces de red del portátil según su localización: el tercer paso consiste en adquirir los parámetros correctos del fichero de configuración del sistema y activar la interfaz de red con estos valores.

A continuación se hará un rápido recorrido por las tecnologías que han sido utilizadas para el desarrollo del proyecto:

- El conjunto de protocolos de comunicaciones TCP/IP [2]. El objetivo principal del proyecto es la configuración de interfaces de redes TCP/IP, por tanto se hará una breve exposición de este conjunto de protocolos.
- Los estándares IEEE 802.3 [3] y 802.11 [4], más conocidos como tecnología Ethernet. La localización, que es el segundo paso básico de nuestra aplicación, se apoya en gran parte en peculiaridades del protocolo Ethernet, perteneciente a la subcapa de acceso al medio. Cada tarjeta de comunicaciones de tipo Ethernet posee una dirección de acceso al medio (MAC), que es unívoca. El mecanismo de localización de la ubicación actual del portátil (segundo paso básico del proyecto) se basa exclusivamente en este hecho: estas direcciones son asignadas por el fabricante de las tarjetas y no pueden ser modificadas a través de software.

La localización consiste en el espionaje del tráfico de la red de área local en la que se encuentra el ordenador portátil (ver siguiente sección) para capturar las direcciones MAC de las máquinas que están conectadas en la red. A partir de estas direcciones, y habiendo configurado correctamente el sistema, este debería ser capaz de determinar donde se encuentra y poder configurar correctamente la interfaz de red. Las direcciones MAC que se especifiquen en el fichero de configuración deberían ser de máquinas que suelen estar normalmente encendidas y generando tráfico de red, como servidores de aplicaciones, servidores web, servidores DNS etc...

- El protocolo de asignación dinámica de direcciones DHCP [5]. Una vez que ya se conoce la localización del ordenador portátil, se procederá a configurar la interfaz de red, extrayendo los parámetros necesarios del fichero de configuración o usando el protocolo DHCP para la obtención de los mismos por la propia red. El usuario especificará las redes en las que se use este tipo de servicio.
- La biblioteca para espionaje de tráfico de red libpcap [6], usada para la fase de localización, como se ha comentado anteriormente.
- El lenguaje XML [7]. Los mecanismos de configuración del sistema están basados en dicha tecnología.
- Las bibliotecas gráficas GTK+ [8]. El proyecto ha sido dotado de una interfaz gráfica GTK+ que pretende hacer el proceso de configuración y el uso del sistema rápidos e intuitivos.
- GNU/Linux [9]. El proyecto ha sido diseñado e implementado para una plataforma GNU/Linux, un sistema operativo multiplataforma tecnológicamente avanzado de la familia UNIX [10].

2. Estudio de trabajos en el mismo área

Antes de la implementación del proyecto TuX-Mobile, se ha realizado un estudio de los paquetes de software disponibles que realizan tareas similares, comprobando que todas ellas dejaban ciertos aspectos y necesidades sin cubrir (ficheros de configuración poco intuitivos, carencia de interfaz gráfica etc...). A continuación se hará una breve descripción de estos paquetes.

2.1. Laptop-net

El paquete Laptop-net [34] ha sido desarrollado en el MIT para portátiles HP-OmniBook sobre Debian GNU/Linux. Debe haber escrita información sobre cómo se configura e inicializa la red para cada localización (IP, máscara de subred, gateway ...). A esa información se la llama Scheme (esquema). Los esquemas pueden también especificar que las direcciones se obtengan median DHCP. En todo momento hay un Scheme seleccionado y existe uno por defecto. Su fichero de configuración (/etc/default/laptop-net) se separa en las siguientes secciones:

SCHEME: El identificador del esquema.

INTERFACE: El nombre de la interfaz de red que se utiliza (eth0, wlan0 ...).

MODULE-NAME: Especifica el nombre del módulo del kernel que contiene el driver para esa interfaz de red. El paquete se encarga de cargar y descargar los módulos.

DHCP: Se activa con el valor "yes", e indica que la dirección se obtendrá por DHCP. Si contiene el valor "no", usará la configuración estática. Las variables para la configuración estática son las comunes: ADDRESS, NETMASK, GATEWAY, BROADCAST, NETWORK, DOMAIN, SEARCH y NAMESERVERS.

Hay información de más alto nivel denominada Profiles que se indexan según las distintas direcciones IP disponibles. En esta información se describirán servicios como las impresoras disponibles en cada red (los Profiles consisten en copias de ficheros de configuración como /etc/printcap). Se definen en el directorio /etc/laptop-net/profiles y cada Profile corresponde a un subdirectorio de éste. Dentro de los subdirectorios pertenecientes a estos perfiles existen más subdirectorios, cada uno dedicado a un propósito general:

patterns: Fichero que contiene los patrones que especifican las direcciones IP pertenecientes a un perfil.

files.d/: Contiene los ficheros que deben copiarse cuando se selecciona un patrón.

Before-select: Fichero que especifica programas que se deban ejecutar antes de seleccionar un patrón.

After-select: Fichero que especifica programas que se deban ejecutar después de seleccionar un patrón.

- deselect: Fichero que especifica programas que se deban ejecutar antes de de seleccionar un patrón.
- rc.d/: En este directorio habrá enlaces simbólicos a los scripts de inicialización para gestionar los servicios del sistema operativo.

De esta forma permite configurar las interfaces de red de los equipos portátiles de una forma más cómoda para el usuario cuando lo usa en diferentes entornos de red, detectando cuando se inserta o se quita un cable de red de la tarjeta (no todos los drivers de tarjetas de red soportan esta opción de notificación cuando se inserta un cable o se desconecta). Una vez que se detecta un cambio de cable (por lo que se supone un cambio de localización), si se está usando DHCP, al adquirir la nueva dirección IP ya se averigua inmediatamente la ubicación actual por que pertenecerá a algún patrón de la configuración y se automatiza la configuración. Si no se usa DHCP, el usuario deberá especificar la nueva dirección IP y a partir de ella se determinará un patrón. Es una solución complicada, y el programa carece de interfaz gráfica de usuario y más métodos de detección de cambios de localización.

2.2. Laptop-netconf

Laptop-netconf [34] es un programa que detecta otras máquinas en la red local en la que nos encontremos y configura de una forma adecuada la interfaz de red dadas unas configuraciones establecidas de antemano. En un fichero de configuración llamado opts se especifican:

- La interfaz de red en el que se debe hacer la detección.
- La configuración de las redes de área local a las que podemos acceder. La detección se realiza enviando paquetes ARP y escuchando las respuestas. Una red se determina cuando en los paquetes ARP de respuesta se identifica una dirección de hardware que conocemos; entonces se ejecuta el script correspondiente al perfil localizado. Sus distintas partes son:
 1. Dirección IP que se debe tomar en esta red: las direcciones IP con las que debe probar para recibir respuestas ARP más la dirección MAC con la que debe coincidir esas respuestas para determinar una ubicación.
 2. El perfil que corresponde a esta ubicación, o sea que script de shell debe ejecutar.

Una vez que se ha determinado en qué red se está viendo qué máquinas se tienen alrededor, se configuran las interfaces ejecutando una serie de scripts de shell que el usuario especifica (que configuran las opciones de red haciendo enlaces simbólicos de los distintos ficheros de configuración alternativos con los del sistema y ejecutando los comandos de configuración `ifconfig`, `route` etc). Es bastante sencillo (un solo fichero escrito en C de 586 líneas) que ofrece una solución simple al problema. Su fichero de configuración también es simple e intuitivo, pero carece de interfaz gráfica y de la posibilidad de usar DHCP.

2.3. Intuitively

Como el paquete anterior, Intuitively [42] configura de una forma automática la interfaz de red descubriendo en que red local se está mediante según las respuestas de los paquetes ARP que emite. En un fichero de configuración se describen las máquinas que posiblemente se encuentren activas en las distintas redes en las que podemos usar el ordenador portátil, normalmente servidores que estén funcionando de una forma ininterrumpida. Intuitively manda por la red falsos paquetes ARP y espera respuesta de esas máquinas para saber en que red local se encuentra y así poder configurar el sistema. Este sondeo se repite hasta un máximo de 6 veces y deja la configuración de red en su estado inicial. Si encuentra y reconoce la red, crea enlaces simbólicos desde la raíz del sistema de ficheros a los archivos que contenga en el directorio `/etc/intuitively/NETWORK` (de esta forma se puede mantener la configuración de impresoras, del sistema apt etc...). Se puede también especificar scripts que se quieran ejecutar cuando se detecte la nueva ubicación. Este paquete se centra más en la localización propiamente dicha, dejando de parte del usuario el que se deba hacer una vez que se ha determinado la ubicación.

2.4. Whereami

El programa Whereami [35] supone que ha habido un cambio de ubicación del portátil cuando el sistema arranca o cuando se inserta una tarjeta de red, y automatiza la configuración de la red que habitualmente se realiza a mano. La tarea de localizar la ubicación actual corresponde a unos pequeños scripts de unas pocas líneas de código que se ejecutan en un orden establecido por el programa. Estos scripts están situados en el directorio `/etc/whereami/tests/` y al ejecutarse deben poder determinar la ubicación, escribirla en un fichero de configuración y activar una variable de entorno que se llama `LOCATED` que se usa como flag dándole un valor de 1. En la documentación no se explica de una forma detallada el proceso de creación de estos scripts, que los debe crear el propio usuario (usando comandos como `ping` para determinar la localización). Los scripts que se deben ejecutar (una vez determinada la ubicación actual) para configurar la red se especifican en el fichero de configuración donde cada ubicación es precedida por un operador (+, -, =) y se relaciona con un script de una manera un tanto críptica:

```
[ operador, ubicación, script ]
```

+: el script se ejecuta si ahora estás en esa ubicación y antes estabas en cualquier otra.

- : el script se ejecuta si no estás ahora en esa ubicación pero sí que estabas cuando se aplicó el script la última vez.
- =: el script se ejecuta si estás en esa ubicación sin preocuparse de cual era tu anterior ubicación.

De los cuatro paquetes destacó por ser el más complicado, con unos ficheros de configuración poco intuitivos. Además, delega en el usuario la creación de sus propios scripts de shell para determinar la localización, centrándose en el problema de qué se debe hacer cuando se está en un sitio determinado y no en el problema de determinar la ubicación del ordenador, justo lo contrario que los dos paquetes anteriores.

2.5. Conclusiones sobre los paquetes existentes

La primera solución, Laptop-net, intenta hacer más cómoda la configuración centrándose más en el qué hay que hacer que en el cuándo, al igual que el paquete Whereami. Las otras dos soluciones (Laptop-netconf e Intuitively) se centran más en averiguar la localización del ordenador en cada momento, las dos usando un método similar, el envío de paquetes ARP, y delegando la configuración del sistema en programas escritos por el propio usuario.

Tras el estudio previo no se ha considerado completo ninguno de estos paquetes. Por esta razón, el proyecto TuX-Mobile pretende cubrir todas estas carencias.

3. Objetivos: El Proyecto TuX-Mobile

El objetivo del proyecto TuX-Mobile es lograr implementar un sistema para la configuración automática de interfaces de red que satisfaga las necesidades de los usuarios. Después de estudiar los programas disponibles que realizaban estos cometidos, se llegó a la conclusión de que ninguno de ellos era completo y de que ninguno presentaba una solución adecuada, sobre todo para el usuario poco experimentado. Los objetivos principales son:

- Hacer que el sistema sea transparente para el usuario respecto a su funcionamiento. El usuario tan sólo debería preocuparse por la configuración del sistema. Una vez que el sistema esté configurado completamente, el usuario no se deberá preocupar de configurar las interfaces de red de su ordenador portátil cada vez que cambie de red de área local.
- Crear una herramienta compatible con todos los entornos GNU/Linux existentes. En la actualidad hay una gran cantidad de distribuciones de este sistema operativo, y un objetivo del proyecto es hacer una herramienta que sea compatible con todos ellos.
- Crear una herramienta completa, que no delegue funciones en el usuario. La sensación que creó el estudio previo de los programas existentes fue que todos ellos delegaban funciones complejas a los usuarios, tales como la creación de scripts para la configuración. Un objetivo del proyecto es crear una herramienta de fácil uso, que no delegue este tipo de cometidos a los usuarios.

- La creación de un sistema fácil de utilizar y de configurar. Este objetivo se pretende alcanzar dotando al sistema de una interfaz gráfica intuitiva.

4. Implementación

Para el desarrollo del proyecto TuX-Mobile se ha seguido un ciclo de vida en espiral [37], por tanto el diseño ha sido refinado en cada iteración de éste. Los prototipos han sido depurados en el ciclo de vida, y en cada iteración se les ha añadido nuevas funcionalidades. El lenguaje de programación que se ha usado para crear la aplicación es el lenguaje C.

4.1. Identificación de entidades externas

Las entidades externas con las que interactúa el sistema TuX-Mobile son:

- Controladores de los dispositivos de red. El programa realiza consultas mediante llamadas al sistema para averiguar el estado de los dispositivos de red, cuyos manejadores o "drivers" se deben encontrar instalados en el núcleo del sistema.
- El sistema TuX-Mobile también se comunica con un programa cliente de DHCP externo a la aplicación. La interacción con este es su invocación cuando sea necesaria y su posterior desactivación cuando no se deba seguir utilizando. El script de invocación del cliente DHCP se llama dhcp.sh.
- Los programas de configuración del sistema ifconfig y route. Estos programas serán invocados desde un script del intérprete de comandos llamado up.sh, que se describirá en un apartado más adelante.

4.2. Identificación de tareas principales

En el sistema se identifican las siguientes tareas principales a realizar:

- Detección periódica del estado de las interfaces de red y posibles cambios de localización.
- Determinación de la ubicación actual. Haciendo un espionaje de los paquetes que viajan por la red en la que estamos y consultando el fichero de configuración podremos determinar la ubicación actual siempre que esta esté representada en dicho fichero de configuración.
- Configuración de las interfaces de red en función de la tarea anterior, o la invocación del cliente DHCP, según esté especificado en su fichero de configuración, llamando a los comandos ifconfig y route adecuadamente.

4.3. Jerarquía de módulos

El sistema está compuesto por una serie de módulos:

- Módulos de Detección de Cambios: estos módulos se encargan de ofrecer una interfaz para la revisión de los distintos dispositivos de red. Existen dos módulos de este tipo, el módulo genérico para dispositivos de red y el específico para dispositivos inalámbricos.
- Módulo de Detección de Ubicación: el módulo que ofrece el servicio de detección de la localización actual del portátil.
- Módulos de Configuración: módulos que se encargan de configurar las interfaces.
- Módulos de la Interfaz Gráfica: son los módulos que se encargan del manejo de la interfaz gráfica de usuario (GUI).

4.4. Fichero de configuración

El fichero de configuración del sistema TuX-Mobile es muy sencillo y comprensible, ya que está basado en XML.

Los elementos que componen el fichero se identifican con la etiqueta "net", y en estos se incluye el nombre de esa red y todos los parámetros necesarios para la configuración de una interfaz de red en una red dada:

name es el nombre de la red.

dhcp especifica si en esa red se usará el protocolo DHCP para obtener los parámetros. Cuando su contenido es "yes" es que sí que se usa, cuando es "no" es que no se usa.

ip especifica la dirección ip que se debe poner en esa red.

mask es la máscara de subred

gateway es el encaminador por defecto que habrá en esa red

dns son los servidores de nombres que se usarán

mac son las direcciones de acceso al medio por las que el programa decidirá que se encuentra en esta red.

En el fichero de configuración habrá tantos elementos "net" como redes queramos tener configuradas en el sistema. El formato del fichero es extremadamente sencillo, por lo que se puede configurar a mano si se desea en vez de usar la interfaz gráfica para ello, aunque siempre será más recomendable que se use la interfaz gráfica puesto que verifica la validez de los parámetros insertados.

4.5. El comando spyMAC

Con el proyecto TuX-Mobile se ha adjuntado un comando que facilita la obtención de las direcciones MAC de una red, ya que esto sería una tarea complicada si el usuario se trasladase a una red en la que no tenga acceso a otros ordenadores y no se pudiera poner en contacto con su administrador. El comando spyMAC se basa, como el paso de localización de TuX-Mobile, en el espionaje del tráfico de la red actual. El comando se dispone a espiar la red que tengamos en el interfaz eth0, y ofrece un listado de hasta 20 direcciones MAC, ya que captura 10 tramas de la red antes de presentar la información.

5. Conclusiones

Todos los objetivos descritos se han cumplido, dando como fruto el sistema TuX-Mobile.

Se consiguió crear el sistema ajustándose a su especificación y se logró usar las bibliotecas elegidas para poder hacerlo. También se logró escribir el programa TuX-Mobile de una forma clara y comprensible, con una jerarquía de módulos razonable y equilibrada. El proyecto TuX-Mobile queda abierto a nuevas versiones en las que se podrían incluir nuevas funcionalidades y aumentar las prestaciones del mismo.

El proyecto ha servido para la familiarización con el entorno UNIX y con la programación en C con bibliotecas gráficas sobre él. Además también ha servido como experiencia en el diseño y desarrollo de soluciones para sistemas y redes, y a manejar las bibliotecas de espionaje de red y de tratamiento de ficheros XML.

Se ha logrado crear una herramienta útil, surgida de la necesidad de hacer el uso de un ordenador portátil en varias redes de área local más fácil para los usuarios de sistemas GNU/Linux. Hemos usado la licencia libre GPL [41] para el proyecto, lo que da derecho a todos los interesados para modificar, optimizar y mejorar el código fuente de la aplicación.

Asimismo se han desarrollado utilidades y herramientas que no entraban dentro de los objetivos del proyecto TuX-Mobile. Se han creado un conjunto de módulos que pueden ser muy útiles para la creación de otras herramientas de gestión de redes. Aparte de esto, también se ha implementado el comando spyMAC de captura de direcciones MAC que puede resultar valioso para la administración de redes de área local y para la implementación de otras herramientas de administración.

El tiempo estimado que se ha dedicado al estudio, análisis de requisitos, diseño e implementación del proyecto y su documentación es de alrededor de 200 horas, repartidas aproximadamente como sigue:

- Alrededor del 10% del trabajo para el estudio de los programas ya existentes descritos en el primer capítulo de esta memoria.
- Aproximadamente un 5% en la creación del documento de especificación de requisitos.
- En torno al 40% del trabajo para el diseño y la creación de los módulos y herramientas aquí descritas.
- Alrededor del 35% para la creación de la interfaz gráfica de usuario con Glade/GTK+ y el manejo del fichero de configuración XML.
- Cerca del 10% en la creación de la documentación y su corrección.

La relación entre el resultado del proyecto y el tiempo que se ha dedicado a la creación del mismo resulta bastante aceptable, más aun habiendo sido alcanzados todos los objetivos que se habían marcado.

La documentación del proyecto, capturas de pantalla y el código fuente están disponible en la siguiente dirección:

<http://gsyc.escet.urjc.es/~esoriano>

6. Referencias

1. E. Soriano. El proyecto TuX-Mobile. <http://gsyc.escet.urjc.es/~esoriano>, 2002.
2. A. S. Tanenbaum. Redes de Computadoras 3a Edición, Capítulos quinto y sexto . Pearson, 1997.
3. A. S. Tanenbaum. Redes de Computadoras 3a Edición, Capítulo cuarto. Pearson, 1997.
4. Grupo de Sistemas y Comunicaciones Universidad Rey Juan Carlos. Redes de área local inalámbricas. <http://gsyc.escet.urjc.es/docencia/assignaturas/redes-I/transparencias/ral-inalambrica/index.html>, 2001.
5. R. Droms. Dynamic Host Configuration Protocol RFC 2131. <http://www.freesoft.org/CIE/RFC/2131/>, 1997.
6. T. Carstens. Programming with pcap. <http://www.tcpdump.org/pcap.htm>, 2002.
7. Extensible Markup Language. <http://www.w3.org/XML/>, 1997.
8. GTK+. <http://www.gtk.org/>
9. Linux Home Page at Linux Online. <http://www.linux.org/>
10. E. Nemeth. UNIX System Administration Handbook 3rd Edition. Prentice Hall PTR, 2000.

11. International Organization for Standardization. <http://www.iso.ch>.
12. A. S. Tanenbaum. Redes de Computadoras 3a Edición, Capítulo primero. Pearson, 1997.
13. E. Nemeth. UNIX System Administration Handbook 3rd Edition, capítulo 16. Prentice Hall PTR, 2000.
14. A. S. Tanenbaum. Redes de Computadoras 3a Edición, Capítulo cuarto. Pearson, 1997.
15. V. Bharghavan, A. Demers, S. Shenker, L. Zhang. MACAW: A Media Access Protocol for Wireless LAN's. ACM SIGCOMM, 1995.
16. J. Deng and Z. J. Haas, Dual Busy Tone Multiple Access (DBTMA): A New Medium Access Control for Packet Radio Networks, IEEE ICUPC'98, Florence, Italy, 1998.
17. Tcpdump Public Repository. <http://www.tcpdump.org>
18. The Internet Engineering Task Force. <http://www.ietf.org>
19. J. K. Reynolds. BOOTP Vendor Information Extensions RFC 1497. <http://www.faqs.org/rfcs/rfc1497.html>, 1993.
20. W3 Consortium. <http://www.w3.org>
21. HyperText Markup Language. <http://www.w3.org/MarkUp>, 1995.
22. M. Bryan. SGML An Author's Guide to the Standard Generalized Markup Language. Addison-Wesley Pub, 1988.
23. D. Veillard. The XML C library for Gnome. <http://xmlsoft.org>
24. GNU's Not Unix! - the GNU Project and the Free Software Foundation (FSF). <http://www.gnu.org>
25. GLADE GTK+ User Interface Builder. <http://glade.gnome.org>
26. The Linux Kernel Archives. <http://www.kernel.org>
27. Portable Operating System Interface(POSIX). ISO/IEC 9945-1:1996, IEEE.
28. Ghostscript-GNU-Project-Free_Software_Foundation(FSF). <http://www.gnu.org/software/ghostscript/ghostscript.html>
29. GCC-GNU-Project-Free_Software_Foundation(FSF). <http://www.gnu.org/software/gcc/gcc.html>
30. GLIB-GNU-Project-Free_Software_Foundation(FSF). <http://www.gnu.org/software/libc/libc.html>
31. Solaris 9 Operating System. <http://www.sun.com/software/solaris>

32. Windows Family Home Page. <http://www.microsoft.com/windows>
33. C. Hansom. The Laptop-net Manual. <http://www.swiss.ai.mit.edu/projects/omnibook/documentation/laptop-net.html>, 2001.
34. Laptop-netconf Package. <http://packages.debian.org/testing/net/laptop-netconf.html>
35. Whereami? - Network Autoconfiguration Utility. <http://mcmillan.net.nz/linux/whereami/>
36. IEEE Recommended Practice for Software Requirements Specifications. IEEE Std 830-1998.
37. B. Boehm. A Spiral Model of Software Development and Enhancement. ACM SIGSOFT Software Engineering Notes, 1986.
38. Document Object Model (DOM). <http://www.w3.org/DOM/>, 2002.
39. The SAX Project. <http://www.saxproject.org/>
40. Wireless Tools for Linux. http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Tools.html
41. GNU General Public License - GNU Project - Free Software Foundation (FSF). <http://www.gnu.org/copyleft/gpl.html>
42. Intuitively. <http://www.samfundet.no/~tfheen/intuitively.htm>