

Enlace Hispanoamericano de Salud: Aplicación del software libre en Cooperación al Desarrollo *

Joaquín Seoane Pascual

joaquin@dit.upm.es

Arnau Sánchez Sala

arnau@gbt.tfo.upm.es

Valentín Villaroel Ortega

valentin@gbt.tfo.upm.es

Andrés Martínez Fernández

andresmf@tsc.uc3m.es

Francisco del Pozo Guerrero

fpozo@gbt.tfo.upm.es

25 de septiembre de 2002

Resumen

Este artículo describe la aplicación de software libre en el programa Enlace Hispano Americano de Salud (EHAS), cuyo objetivo es contribuir a la mejora del sistema público de asistencia sanitaria en zonas rurales aisladas de países de América Latina, por medio de las comunicaciones y la informática. Dicho programa investiga, desarrolla e implanta soluciones de comunicación de bajo costo y adaptadas a necesidades extremas, utilizando y desarrollando software libre allá donde es posible.

1. Introducción

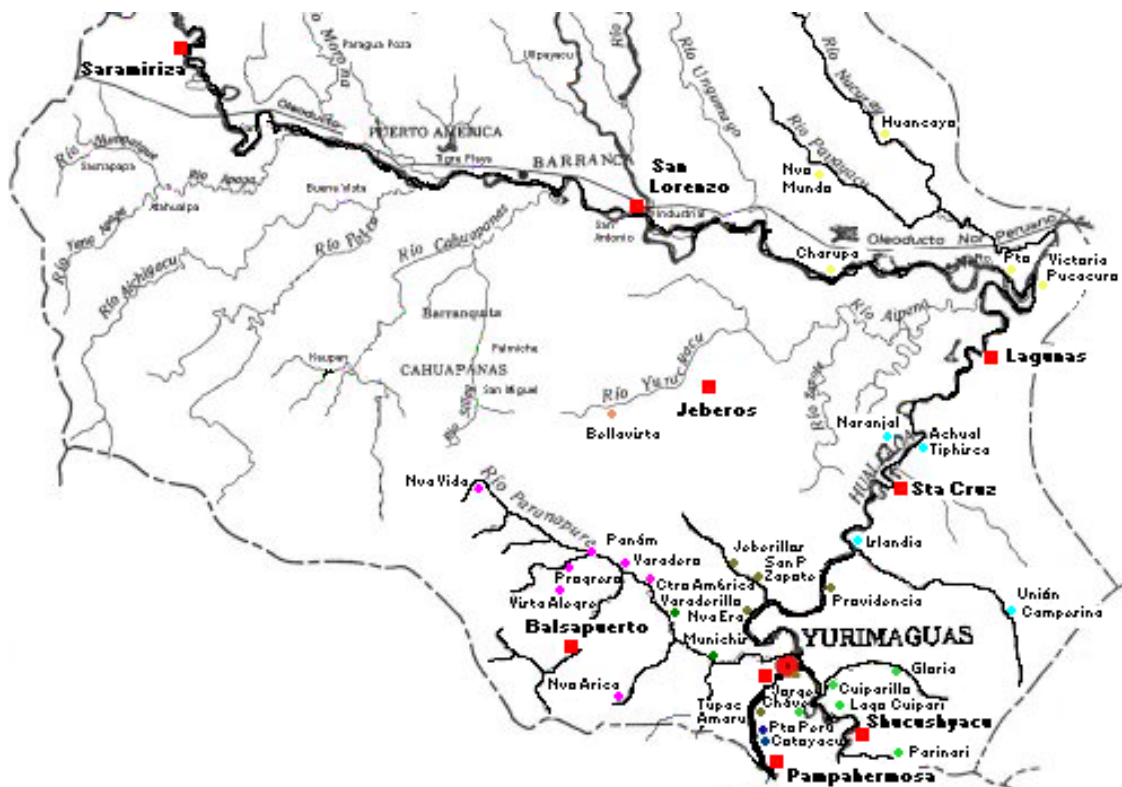
Se han dado numerosas razones para utilizar y desarrollar software libre en países en desarrollo y, en particular, en proyectos de desarrollo, pero, la realidad indica que su aplicación es aún limitada, por diversas razones, entre las que cabe destacar el bajo nivel educativo en informática, el monopolio de formatos en ofimática, y el recurso masivo a la copia ilegal. Como en todas partes, su introducción empieza allá donde su uso es, sin posibilidad de discusión, la mejor solución técnica y económica.

* Copyright (C) 2002 Joaquín Seoane Pascual, Arnau Sánchez Sala, Valentín Villaroel Ortega, Andrés Martínez Fernández y Francisco del Pozo Guerrero.

Este es el caso de las soluciones que desarrolla el programa Enlace Hispano Americano de Salud (EHAS[15]), liderado por el Grupo de Bioingeniería y Telemedicina de la Universidad Politécnica de Madrid, en su vertiente de investigación y desarrollo, y por la ONG Ingeniería Sin Fronteras, en su vertiente de realización de proyectos. Cuenta con la participación de universidades e instituciones públicas en Perú, Colombia y Cuba y es apoyado financieramente por el Plan Nacional de I+D+I, Colciencias, Concytec, AECI, CYTED, el programa InfoDev del Banco Mundial, y Comité de Cooperación y Solidaridad de la Universidad Politécnica de Madrid, entre otros.

El programa EHAS parte del estudio de necesidades de personal sanitario en regiones aisladas de Perú y Nicaragua, así como de sus condicionantes en comunicaciones y energía. Si bien cada zona estudiada es diferente, el mayor problema es la incomunicación física, ya que en algunos casos el técnico sanitario puede llegar a necesitar varios días de viaje para acceder a su centro sanitario de referencia, del que depende. Sin llegar a intentar resolver el problema físico, la telecomunicación es el vehículo obvio para paliar el aislamiento personal y profesional, pudiendo facilitar las consultas profesionales, la formación, el intercambio de informes, la coordinación de emergencias, el pedido de medicamentos, etc. Las razones para no disponer de un servicio de telecomunicaciones apropiado son el desinterés de las operadoras en las zonas aisladas, el escaso presupuesto de los puestos de salud, la baja formación de los técnicos sanitarios, especialmente en relación con el uso de la informática, la ausencia de una fuente de energía eléctrica apropiada, y condiciones ambientales extremas.

Tras establecer las condiciones necesarias para llevarlo a cabo, entre los años 2000 a 2002 se ha puesto en marcha un proyecto piloto en la provincia de Alto Amazonas del departamento de Loreto en Perú, con objeto de demostrar una solución de bajo coste y evaluar su impacto. Dicho proyecto involucra al Hospital Provincial de la capital, Yurimaguas, y a 40 establecimientos de salud de dos categorías: centros de salud y puestos de salud. Los centros de salud están dotados de varias personas, al menos una de ellas médico, y suelen tener teléfono y grupo electrógeno, funcionando algunas horas al día. Los puestos de salud dependen de los anteriores y sólo cuentan con una persona de formación limitada y carecen de teléfono y cualquier fuente de energía. El mapa que sigue muestra el área de intervención, con los centros marcados con cuadrados rojos y los puestos como puntos de color.



La experiencia obtenida con este proyecto ha dado pie para definir otros nuevos en distintos escenarios, con la misma tecnología mejorada o con otras distintas. A continuación se describen las tecnologías utilizadas, haciendo hincapié en el software libre empleado, modificado o construido.

2. Medios de comunicación

En zonas como Alto Amazonas, con establecimientos separados decenas de kilómetros, sin teléfono, la infraestructura más sencilla y usada para comunicar los puestos de salud con su centro de salud es la radio VHF, ya que su alcance puede ser del orden de 50 Km, limitado por la potencia de transmisión y la altura de las antenas, que deberán compensar la curvatura de la tierra y salvar los obstáculos, aunque tiene bastante tolerancia a los mismos. Se eligen pues en estos casos radios VHF convencionales que se utilizan normalmente para voz, pero que, intermitentemente, pasan a intercambiar datos entre un ordenador cliente en el puesto de salud y un servidor situado en su centro de salud de referencia. Una instalación solar en el puesto de salud alimenta la radio y el ordenador cliente, así como otros aparatos auxiliares (modem radio, luminarias, etc). Los centros de salud disponen de un cargador de baterías para alimentar el servidor las horas necesarias, aunque el grupo electrógeno esté apagado, ya que sólo se enciende a la puesta del sol, durante unas cuatro horas. El servidor a su vez se comunica intermitentemente con internet a través del teléfono o, en algunos casos, a través de un terminal VSAT. En las fotos siguientes se ven las antenas de VHF de un centro de salud y un puesto de salud, éste con su placa solar.





En zonas muy aisladas, VHF no se puede utilizar. En estos casos puede utilizarse

HF (onda corta) por reflexión ionosférica, salvándose distancias de centenares o miles de kilómetros. Lamentablemente en HF tenemos enlaces de peor calidad y con mucha variabilidad en cortos intervalos de tiempo, además de que sólo puede usarse a ciertas horas, dependiendo del canal, y con protocolos y modulaciones especiales. Luego examinaremos brevemente las soluciones propuestas para HF.

Otra solución explorada para zonas muy aisladas ha sido la utilización de satélites de baja órbita (LEO). Estos satélites no son geoestacionarios, pero pueden utilizarse cuando pasan por encima de un punto para intercambiar ficheros con muy baja energía. Los principales inconvenientes son que no pueden utilizarse para voz y dependen de microsátélites compartidos de vida incierta. Ciertamente podría usarse la moderna telefonía satelital en la mayoría de los lugares, pero los costes actuales resultan excesivos.

Finalmente también se está explorando la aplicabilidad de la tecnología de redes locales inalámbricas en microondas (IEEE 802.11b) en diversos escenarios. Con las regulaciones vigentes en Hispanoamérica se pueden conseguir enlaces de decenas de kilómetros a potencias muy bajas, con un ancho de banda mucho mayor que las soluciones anteriores, lo que abre el camino a aplicaciones muy diferentes. Su mayor inconveniente es la necesidad de que no haya obstáculo alguno entre los puntos a comunicar, algo muy caro de conseguir en selva, por ejemplo. No obstante se está explorando la posibilidad de construir repetidores activos autónomos de bajo costo, lo que facilitaría su aplicación en ese ámbito.

3. Correo electrónico

Excepto en el caso de las redes inalámbricas en microondas, vemos que las soluciones propuestas se basan en conexiones intermitentes de bajo ancho de banda y utilizando medios diversos. Por ello todas las aplicaciones contempladas se construyen a través de correo electrónico, ya sea interpersonal, ya entre programas. Así los mensajes pueden atravesar distintos medios (radio VHF o HF, conexiones telefónicas a veces de mala calidad, o almacén intermedio en satélites LEO) sin problemas.

4. Microrredes VHF

Llamamos microrred VHF a una red formada por varios puestos de salud que dependen de un centro de salud, en el cual se dispone de acceso a Internet, ya sea intermitente (RTC) o permanente (VSAT). La comunicación entre puestos y centro es por medio de las radios VHF que también se utilizan para voz. En principio sólo es necesaria una interacción cliente servidor entre los puestos y el centro, y este hecho permite reducir el coste de las infraestructuras necesarias, especialmente las instalaciones aéreas (conjunto torre-antena) y energéticas (alimentación solar), como luego veremos.

En Alto Amazonas las condiciones locales impusieron que los clientes fueran máquinas portátiles MS-Windows, lo que condicionó la solución adoptada y ocasionó no pocas dificultades, debido al carácter cerrado de la plataforma y del soft-

ware auxiliar, así como por lo limitado de la arquitectura y su vulnerabilidad. La foto muestra el uso de uno de esos clientes.



Los servidores en cambio fueron máquinas Debian GNU/Linux realizadas con componentes PC/104[12], de bajo consumo y gran robustez y fiabilidad. La foto muestra el interior de uno de estos servidores, con su radio y modem.



Para comunicar vía radio clientes y servidores se utiliza el protocolo AX.25[1], una adaptación de X.25 realizada por radioaficionados y de la que existe un par de implementaciones libres para Linux y varias implementaciones cerradas, aunque gratuitas en nuestra aplicación, para Windows.

Como las aplicaciones de correo para Windows utilizan protocolos POP y SMTP directamente, fue preciso implementar estos servicios de alguna manera sobre AX.25. En general, sea cual sea la plataforma, parece buena idea implementar protocolos de aplicación estándar de Internet, ya que de ese modo todos los programas diseñados para internet funcionarán directamente. Sin embargo, esto ocasiona dificultades e ineficiencias, por lo que para estaciones de trabajo Linux preferimos utilizar el venerable uucp de Ian L. Taylor[5], paquete libre que ya usamos sobre TCP para intercambiar correo entre los centros de salud y el conmutador central, situado en la Universidad Católica de Lima para el piloto de Alto Amazonas. Dicho paquete soporta diversos protocolos para distintos tipos de conexiones, permite reanudar transferencias abortadas, y es eficiente, pudiéndose montar sobre él paquetes comprimidos con bsmtp[10].

4.1. TCP sobre AX.25

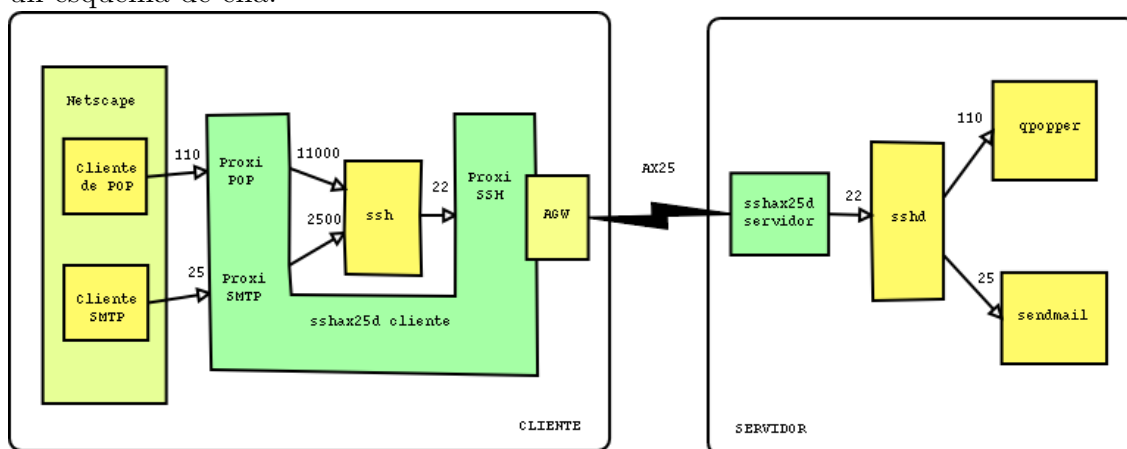
La solución tradicional para aplicaciones de internet, implementada en Linux y Windows, es utilizar AX.25 como nivel de red sobre el que se envían paquetes IP. Esta solución es ineficiente en ancho de banda y no funciona bien, debido al desconocimiento por parte de TCP de un medio semiduplex de elevada tasa de error, latencia y probabilidad de colisión, lo que ocasiona repeticiones innecesarias de paquetes y falsas detecciones de congestión.

Está claro que hay que impedir que TCP intervenga como protocolo de transporte y delegar la responsabilidad de gestión del enlace en AX.25, mucho mejor adaptado para ello. Lo que se hace es engañar a las aplicaciones, haciéndolas hablar con sendos proxis locales de SMTP, POP o lo que sea necesario. Las conversaciones

con esos proxis se trasladan al servidor por medio de AX.25, donde escuchan sus proxis complementarios, que trasladan la conversación a los servidores.

Para multiplexar las distintas conversaciones de aplicación no se utiliza la posibilidad de AX.25 de soportar varias conversaciones, sino que se utiliza SSH[7], protocolo usado habitualmente en Internet para establecer comunicaciones cifradas. El hecho de usar SSH resuelve de una vez tres problemas: la multiplexión de conexiones, el uso eficiente del escaso ancho de banda disponible por medio de la compresión de datos, y la protección de información que puede ser confidencial por medio del cifrado.

Las aplicaciones en el cliente se configuran apuntando a un proxy local de SMTP y POP (puertos 25 y 110). Este proxy detecta las conexiones, lanza un cliente de SSH escuchando en puertos trasladados (2500 y 11000) y redirige los datos de los puertos originales a los trasladados. El cliente de SSH multiplexa, comprime y cifra los datos que recibe y los manda a otro proxy local escuchando en el puerto de SSH (22). Su función es repetir por el canal AX.25 todos los datos de la comunicación SSH. En el servidor otro proxy similar intercambia datos entre el socket AX.25 y otro TCP, donde está el servidor SSH demultiplexor. Fontanería complicada, pero eficiente y extensible a cualquier aplicación cliente/servidor basada en TCP. La figura muestra un esquema de ella:



4.2. Radios y modulaciones

Con radios comerciales de voz para VHF y modulación FM, canalizadas a 25 KHz, se pueden conseguir 9600 bps, pudiéndose doblar la velocidad o reducir a la mitad la canalización si las radios y los modems fueran de muy buena calidad. Ciertamente se pueden utilizar velocidades mayores con otras radios, a costa de mayor ancho de banda, pero no es fácil de conseguir licencia para ello.

Como modems se han utilizado TNC, pequeños ordenadores utilizados por los radioaficionados, diseñados para trabajar en modo terminal y adaptados para comunicaciones entre ordenadores, pero por razones de coste, flexibilidad y fiabilidad se ha optado finalmente por utilizar tarjetas de sonido, usando el procesador central como procesador de señal. Eso nos permite además optimizar las modulaciones, como veremos que se ha hecho en HF. Para radios de voz VHF la modulación más

eficiente utilizable es una FSK G3RUH [2], definida por James Miller y ampliamente implementada en hardware y software, incluido el paquete soundmodem de Thomas Sailer[4].

4.3. Protocolo DAMA

Los clientes y el servidor de una microrred compiten por la única frecuencia disponible, que usan tanto en transmisión como en recepción. Para controlar el acceso al medio, en el entorno de radioaficionado se utiliza normalmente CSMA-CA: una estación escucha si alguien está hablando y, si es así, se espera un tiempo pseudoaleatorio para intentarlo de nuevo, tanto más largo cuantas más veces se haya detectado portadora. La probabilidad de colisión y destrucción de paquetes puede ser muy grande debido a que el tiempo de conmutación entre recepción y transmisión de la radio semiduplex es considerable y durante ese tiempo no es posible escuchar. Además en radio no es posible detectar la colisión y anular la transmisión, como se hace en ethernet, por lo que debe ser el propio protocolo quien, al expirar los plazos, envíe peticiones de retransmisión, con la consiguiente merma en la eficiencia. Además CSMA-CA exige que todas las estaciones se escuchen entre sí, lo que implica torres más altas para que las antenas se vean todas entre sí. Esto es costoso y exige potencia suficiente en las radios para llegar a todos los puntos. Sin embargo en nuestra aplicación sólo hay comunicación cliente servidor.

Una solución es la implementación, al mismo nivel de AX.25, de un protocolo de control de acceso múltiple asignado bajo demanda (DAMA). Dichos protocolos permiten un maestro la asignación de turnos a los esclavos, que deben solicitar el acceso al medio al maestro y usarlo solamente cuando éste lo pida. No existe ninguna implementación de DAMA maestro para Linux y tampoco existía en aquel momento DAMA esclavo para la implementación AX.25 elegida para Windows (AGW, de George Rossopoulos[8], quien lo ha añadido en la última versión de su programa). Debido a que el carácter cerrado de AGW impedía cualquier modificación, se optó por no implementar un maestro DAMA genérico para Linux, sino algo que funcionara correctamente con clientes que desconozcan que se está utilizando acceso bajo demanda, haciéndoles creer que el servidor está ocupado durante la rodaja de tiempo que no les corresponde. La modificación realizada en el núcleo de Linux permite al servidor la asignación de turnos a los clientes que simultáneamente han establecido una conexión con él, sin violar el protocolo AX.25 ordinario.

5. Redes en HF

Con HF ya no son necesarias microrredes, ya que generalmente las estaciones clientes pueden alcanzar directamente un servidor en una capital bien conectada. Sin embargo el canal HF tiene características que hacen difícil trabajar con él, por lo que los modems de HF son extraordinariamente caros, o muy lentos (típicamente de 100 a 300 bps para los de radioaficionados). Para aprovechar el escaso espectro disponible, los canales suelen ser de 3 KHz y la modulación en banda lateral única,

mucho menos robusta que la FM y sometida además a desvanecimientos ocasionados por las incertidumbres de la propagación ionosférica.

Gran parte del trabajo desarrollado para VHF sirve para HF (por ejemplo el mecanismo DAMA de control de acceso al medio en AX.25). Sin embargo la mala calidad del canal hizo necesario trabajar más en profundidad la modulación, así como mejorar el protocolo AX.25 para soportar mejor muchas pérdidas de paquetes. La ausencia de fuentes para Windows determinó que la estación de trabajo fuera Linux, pudiéndose intervenir en modificaciones incompatibles con el estándar AX.25 y en los modems de tarjeta de sonido.

La modificación más importante se hizo sobre un modem (`newqpsk`) añadido por Tomi Manninen al paquete de modems para tarjeta de sonido en espacio de usuario de Thomas Sailer (`soundmodem`). Dicho modem, basado a su vez en otro de Pawel Jalocha, usa 15 portadoras moduladas en DQPSK, entrelazado de bits para luchar contra los errores de ráfagas, y códigos autocorrectores. En él se sustituyó el código autocorrector por turbocódigos convolucionales[3], adaptando y optimizando un diseño de Mathys Walma, además de otras modificaciones, consiguiéndose alcanzar velocidades alrededor de los 2000 bps, según la calidad del canal. Como se ve, este módulo es un ejemplo de la conveniencia del modelo de desarrollo de software libre.

Otra modificación importante fue la retransmisión selectiva de paquetes en AX.25, algo muy necesario en un entorno tan sensible a errores. La modificación introducida no es compatible con el estándar, ya que permite enumerar exactamente qué paquetes se han recibido mal, además de indicar el último correctamente recibido. De esta forma, sólo se retransmiten los necesarios con ventanas de transmisión que siempre contienen el máximo número de paquetes posible.

Finalmente se configuró `uucp` para funcionar directamente sobre AX.25, a través del protocolo y, apto para canales semiduplex libres de errores. Por encima, el agente de transferencia de correo (`postfix`, de Wietse Venema[9]) se configuró para usar un transporte BSMTP sobre `uucp`. BSMTP crea paquetes comprimidos a partir de varios mensajes, que son enviados como ficheros de tamaños parecidos.

6. Satélites de baja órbita

También se ha contemplado el posible uso de microsátélites de baja órbita, que son visibles varias veces al día durante unos minutos, momento que se puede aprovechar para intercambiar mensajes con ellos. Estos satélites también fueron diseñados por radioaficionados, aunque existen satélites que usan la misma tecnología con fines comerciales y de cooperación al desarrollo (Vitasat, Healthsat). Suelen tener transmisores en la banda de UHF y receptores en VHF operando protocolos de difusión y transferencia de ficheros montados sobre AX.25 y conocidos como protocolos PACSAT[6].

En este caso lo que se hizo fue crear un agente de transporte para correo electrónico utilizando los protocolos PACSAT, y conectándolo adecuadamente a un agente de transferencia (en nuestro caso a `sendmail` de Allman[11], pero vale para cualquier otro). Para ello se extrajeron porciones de un programa libre interactivo

llamado pb-pg[14] para extraer las porciones relativas al protocolo.

Uno de los problemas de este sistema es la necesidad de apuntar antenas móviles y la de corregir la frecuencia de la radio para compensar el desvío ocasionado por el efecto Doppler. Además del coste de los equipos controlados, es necesario un sistema de previsión de órbitas. Los mismos satélites envían periódicamente esas previsiones, pero es mucho más conveniente no tener que usarlas, para lo que se han desarrollado antenas fijas y modificaciones de radios.

7. Las aplicaciones

La infraestructura montada con los componentes antes descritos, acompañada de algo de organización y mucho de formación, sirve ya para mejorar la calidad asistencial de los establecimientos de salud aislados. Sin embargo es conveniente desarrollar aplicaciones que den valor añadido a las redes construidas. Por ejemplo, puede ser muy útil facilitar la cumplimentación de informes epidemiológicos que puedan servir al sistema de salud para reaccionar adecuadamente a los brotes epidémicos.

Actualmente se está desarrollando en el piloto de Alto Amazonas un programa de formación en el que la Universidad Peruana Cayetano Heredia, contraparte médica del proyecto en el Perú, prepara cursos que se dividen en lecciones que se envían por correo electrónico como mensajes en HTML con imágenes. Este método, si bien no requiere ninguna herramienta adicional, es poco conveniente para editores y alumnos, por lo que para futuros proyectos del programa EHAS se están desarrollando una serie de herramientas que nos permiten definir cursos en XML de forma abstracta, transformándolos con posterioridad por medio de hojas de estilo XSLT, con ayuda de motores XSLT libres como Saxon[13].

8. Conclusiones

El software libre nos ha permitido desarrollar rápidamente con pocos recursos un conjunto de soluciones de bajo coste para mejorar las condiciones de vida en zonas aisladas y desfavorecidas. También ha ayudado el carácter abierto de la comunidad de radioaficionados. En los casos en que se ha tenido que usar software comercial, las dificultades han sido mayores, ya que se han tenido que soslayar los problemas que su carácter cerrado impedía resolver más directamente. Esperamos que el trabajo realizado y por venir ayude a los sistemas públicos de salud de países en desarrollo, algunos de los cuales son plenamente conscientes de la importancia estratégica del software libre, a desplegar soluciones económicas, eficaces y controladas por ellos.

9. Bibliografía

1. William Beech, Douglas Nielsen y Jack Taylor. AX.25 link access protocol for amateur packet radio v2.2. Technical report, Tucson Amateur Radio Corpo-

- ration, 1997.
2. James Miller. 9600 baud packet radio modem design. <http://www.amsat.org/amsat/articles/g3ru> 1994.
 3. William E. Ryan. A turbo code tutorial. <http://www.ece.arizona.edu/~ryan/turbo2c.pdf>.
 4. Thomas Sailer. Multiplatform soundcard packet radio modem. <http://www.baycom.org/~tom/ha>
 5. Ian Lance Taylor. Taylor uucp 1.06.1. <http://www.airs.com/ian/uucp.html>.
 6. Jeff Ward. AMSAT-NA microsats - protocols. <http://www.amsat.org/amsat/sats/nk6k/msatpro> 1995.
 7. T. Ylonen. The SSH (Secure Shell) remote login protocol. Technical report, Helsinki University of Technology, 1995.
 8. AGW Packet Engine, <http://www.raag.org/sv2agw/agwpe.htm>
 9. Wietse Venema, The Postfix Home Page, <http://www.postfix.org>.
 10. Roland Rosenfeld, Batched SMTP mailer for sendmail and postfix, <http://www.spinnaker.de/del>
 11. Sendmail Consortium, Sendmail Home Page, <http://www.sendmail.org>.
 12. The PC/104 Consortium, PC/104 Embedded PC Modules, <http://www.pc104.org>.
 13. Michael Kay, The Saxon XSLT Processor, <http://saxon.sourceforge.net>.
 14. Bent Bagger, PB and PG for Linux, <http://ibiblio.org/pub/Linux/apps/ham/pbpg-2.0.tar.gz>.
 15. Programa EHAS, <http://www.ehas.org>.