

10. ANEXO I. Compilación y ejecución de OperAIT.

En este anexo se describen los pasos necesarios para compilar y ejecutar la aplicación OperAIT en un sistema Linux.

Para el desarrollo del Proyecto de Fin de Carrera se utilizó SuSE Linux 7.2 corriendo sobre un IBM PC con procesador Intel Pentium (también se probó en Intel 486 DX2) y un núcleo 2.4.16. Los únicos requisitos para compilar y ejecutar la aplicación son el uso de un núcleo 2.4.X, la instalación de la biblioteca de procesamiento XML `expat` y la instalación del código fuente del núcleo sobre el que se ejecutará el programa.

La biblioteca `expat` está actualmente incluida en todas las distribuciones de Linux o, alternativamente, el código fuente puede obtenerse de la página web:

`http://sourceforge.net/projects/expat/`

10.1. Compilación.

Todo el código fuente se encuentra contenido en el directorio `source`, del cual cuelgan cuatro directorios:

- `adk-source`: Este directorio contiene todos los archivos referentes a la biblioteca CAPI ADK de AVM modificada. Los archivos binarios con el código objeto producto de la compilación se almacenarán en el subdirectorio `objs`.
- `c20lib`: Este directorio contiene la biblioteca `c20lib`, que es necesaria para que la AVM CAPI ADK pueda ser compilada.
- `module`: En este directorio se encuentran todos los archivos fuente para el módulo del núcleo `rdsi.o` y la biblioteca `capifuncs.o` (la parte de emulación de CAPI que corre en espacio de usuario). Los ficheros binarios se almacenan en este directorio tras la compilación. Existe un directorio llamado `kernel` que contiene los ficheros que hay que copiar dentro del árbol de directorios que contiene el código fuente de Linux. La ruta exacta es `/usr/src/linux/drivers/isdn/hisax` salvo para el archivo `isdnif.h`, que debe copiarse en el directorio `/`

OperAIT. Operadora del Área de Ingeniería Telemática.

`usr/src/linux/include/linux`. El directorio `module` contiene también dos archivos de proceso por lotes para cargar y descargar el módulo `rdsi.o`.

- `project`: Este directorio almacena el código fuente de la aplicación OperAIT. El fichero ejecutable del programa se almacenará en el subdirectorio `exe`. Existen dos subdirectorios, de nombre `conf` e `isdn2wav`, que almacenan el código fuente y los ficheros objetos tras la compilación de los módulos de configuración y transcodificación WAV/RDSI.

Para poder ejecutar el programa es necesario compilar primero el módulo `rdsi.o` y posteriormente la aplicación OperAIT.

Para compilar los módulos del núcleo, es necesario primero copiar los archivos del directorio `kernel` en el árbol de directorios que contiene el código fuente de Linux. Entonces habrá que compilar el módulo `hisax.o` (ver manuales de compilación del núcleo para ello) y copiar el fichero objeto al directorio `module`²¹. Posteriormente habrá que ejecutar el comando `make` dentro del directorio `module`, para compilar el módulo `rdsi.o`.

La compilación de la aplicación OperAIT se realiza ejecutando el programa `make` dentro del directorio `project`. Esto forzará la compilación de las bibliotecas AVM CAPI ADK y `c20lib`, y de los módulos `isdn2wav.o` y `conf.o`, enlazándose todos ellos al final para formar el programa `operait`.

10.2. Ejecución.

Una vez todo haya sido compilado habrá que cargar los módulos `rdsi.o` y `hisax.o`, en este orden. Esto puede hacerse utilizando el comando `insmod`²² o los archivos de lotes *Carga y Descarga* que están en el directorio `module`. Para que el módulo `rdsi.o` pueda ser cargado, es necesario que exista el fichero asociado al dispositivo `/dev/capi20`. Si éste no existiera, puede crearse con el comando `mknod` o con la utilidad `MAKEDEV` existente en algunas distribuciones Linux.

Para ejecutar el programa es necesario primero crear un archivo de configuración válido para el mismo. El capítulo 11 explica cómo crear dicho archivo de configuración. Este archivo debe situarse en el directorio donde esté el fichero ejecutable del programa o en el directorio `/etc`. Una vez hecho esto, se cambiará el directorio de trabajo a aquél donde esté el fichero ejecutable y se iniciará el programa mediante el comando `./operait`. La versión

²¹ Esto último sólo es necesario si se desea utilizar los archivos de proceso por lotes para cargar y descargar el módulo.

²² El módulo `hisax.o` toma dos parámetros, que pueden variar de equipo en equipos. Ver la documentación del módulo para determinar el valor de los mismos si se utiliza en otro equipos o se quiere cargar el módulo con `insmod`.

actual del programa depende de rutas relativas, por lo que no se debe intentar ejecutar el mismo desde otro directorio.

Para el funcionamiento del programa es necesario que los módulos estén cargados y que el usuario que ejecuta el mismo tenga permisos de lectura y escritura en el dispositivo `/dev/capi20`. Además, el módulo `rdsi.o` sólo admite una aplicación utilizando el dispositivo por lo que, si ya hay una instancia del programa ejecutándose, éste no arrancará.

10.3. Finalización.

El programa no se ejecuta como un servicio, por lo que para terminar el mismo puede pulsarse la combinación de teclas control-C. Si se desea ejecutarlo como un servicio, puede utilizarse el operador `&` para que se ejecute en segundo plano. En tal caso, para finalizar el programa basta enviar la señal `TERM` al `PID` asociado al mismo, mediante el comando `kill`.

10.4. Reconfiguración.

Puede cambiarse la configuración del programa sin necesidad de para el mismo. Para ello se enviará la señal `HUP` al `PID` asociado al programa, utilizando de nuevo el comando `kill`.

