



Universidad Nacional de La Plata

Lector de Noticias

TRABAJO DE GRADO

Mariel Flores

Flavia Piatti

INDICE

INDICE	2
INTRODUCCION	5
OBJETIVO DEL PROYECTO.....	6
INVESTIGACIONES PRELIMINARES.....	7
MARCO TEORICO	7
USENET	7
ORGANIZACION DE USENET.....	7
TRABAJAR CON LA COMUNIDAD USENET	9
Como comunicarse.....	9
Como postear	10
Que no postear	15
Donde postear.....	16
DISTRIBUCION DE ARTÍCULOS EN USENET	16
CONSIDERACIONES SOBRE GRUPOS DE NOTICIAS.....	17
Flamees.....	17
Flames y FAQ	18
Surfeando y espiando	18
COMPORTAMIENTO DE LOS GRUPOS DE NOTICIAS	19
CREACION DE UN GRUPO DE NOTICIAS	19
La discusión	19
La votación.....	20
El resultado.....	21
FORMATO DE LOS ARTÍCULOS.....	22
FORMATO GLOBAL DE UN ARTICULO.....	24
FORMATO DEL HEADER	25
Headers obligatorios.....	25
Headers opcionales.....	27
FORMATO DEL CUERPO	30
FORMATO DE LA FIRMA.....	31
MENSAJES DE CONTROL.....	31
MIME.....	33
Headers.....	33
Tipos de contenido	34
Codificaciones de transferencia de contenido.....	36
LECTORES DE NOTICIAS.....	37
FUNCIONALIDADES PRINCIPALES.....	37
CONTROL DE LOS ARTÍCULOS QUE SE LEERAN.....	38

NN NEWSREADER	40
Historia.....	40
Características.....	40
Comienzo	40
Modos	41
Seleccionando artículos.....	41
Búsquedas por Subjects, Autores y Mensajes.....	42
Leyendo artículos	43
Navegando	44
Obtener Ayuda	45
Emergencias	45
Decodificando artículos.....	46
Posteando artículos	47
Enviando un mail	49
Killiar y seleccionar artículos.....	50
Personalizar nn.....	51
Archivos Internos.....	52
Finalización de Nn.....	53
TIN NEWSREADER	53
Historia.....	53
Características.....	53
Comienzo	54
Modos	54
Seleccionando artículos.....	55
Busqueda a través de Subject	57
Leyendo artículos	58
Navegando	58
Obteniendo ayuda	59
Emergencias	59
Decodificando artículos.....	60
Postear artículos.....	60
Enviando mail	61
Killing y selecting article.....	62
Finalización	62
FREE AGENT	63
Historia.....	63
Características.....	63
Comienzo	64
Modos	65
Interface.....	65
Búsquedas por Subjects, Autores y mensajes.....	67
Seleccionando artículos.....	68
Leyendo artículos	69
Navegando	69

Obtener Ayuda	71
Decodificando artículos	71
Posteando artículos	71
Enviando mail	72
Killiar y purgar	73
Personalizar	73
Archivos Internos.....	74
Multitarea.....	74
Finalización	75
CARACTERISTICAS A CUMPLIR POR EL LECTOR DE NOTICIAS	76
AMBIENTE DE TRABAJO	78
JAVA.....	78
AMBIENTES DE PROGRAMACION JAVA	79
BORLAND JBUILDER.....	80
Entorno de desarrollo	80
Creación de un programa de Java	83
Asistentes	83
Los componentes y la librería.....	84
Acceso a archivos	85
Los JavaBeans	85
Depurador de Jbuilder	88
APLICACION LECTOR DE NOTICIAS.....	90
ARQUITECTURA.....	90
COMPONENTES PRINCIPALES.....	91
Cliente NNTP Y SMTP	92
Grupo de Noticias.....	96
Artículos	97
Usuario	100
Lector de Noticias.....	101
Editor.....	105
FUNCIONALIDAD DE LA APLICACIÓN.....	112
Características.....	112
Comienzo	113
Interface.....	115
Navegando	116
Seleccionando y leyendo artículos.....	117
Modos	119
Posteando artículos	119
Enviando E-mail	120
Búsquedas por Subjects o Autores.....	120
Personalizar	121

Utilización del Editor.....	122
Tratamiento de Direcciones.....	126
Finalización.....	128
USABILIDAD DEL PRODUCTO LECTOR DE NOTICIAS.....	129
CONCLUSION.....	130
NOCIONES BASICAS.....	131
BIBLIOGRAFIA.....	134

INTRODUCCION

En estos días donde el mail se ha vuelto una herramienta de comunicación que compite con el teléfono en los lugares de trabajo y hogares, donde la gente está comenzando a acceder a Internet en gran número, no podemos dejar de mencionar y analizar el fenómeno llamado USENET, como una nueva opción de comunicación.

USENET, Net News o News son términos equivalentes para un mismo concepto, es una red de noticias mundial con sitios de computadoras interrelacionadas, que intercambian mensajes públicos de usuarios, llamados artículos. Estos artículos son una mezcla ecléctica de preguntas, comentarios e ideas de todo tipo, expresadas para la vista de miles de participantes de todos los sitios. Estos artículos se encuentran ordenados dentro de categorías llamadas newsgroups (grupo de noticias).

Los newsreaders (lectores de noticias) son programas que permiten leer newsgroups. Los mismos facilitan a los usuarios la tarea de acceder a los grupos que son de su interés y almacenan determinada información como cuales artículos fueron leídos y a que newsgroups se ha suscrito el usuario.

El fenómeno Java ha cautivado la imaginación de los programadores de todo el mundo y encara la nueva era del desarrollo de aplicaciones distribuidas. El atractivo de Java radica en su simplicidad, familiaridad y cuidadosa selección de características que incluye y excluye.

Nuestro trabajo se basa en el desarrollo de un Lector de Noticias implementado como una aplicación Java, con interfaz iconizada que integra, de forma simple, la distribución de mails y artículos dentro de USENET. Lo hemos realizado a través del entorno Borland Jbuilder, lenguaje de desarrollo de Aplicaciones Java.

OBJETIVO DEL PROYECTO

La idea principal de nuestro proyecto es la implementación de un Lector de Noticias online/offline, desarrollado en lenguaje Java.

Para llevar a cabo nuestro desarrollo tuvimos como objetivo de implementación el cumplimiento de las características básicas de la mensajería dentro de Usenet, la integración de dos importantes servicios de internet, mail y newsreader y la implementación de una interfaz gráfica que permita visualmente guiar al usuario por las distintas funcionalidades del lector. Asimismo hicimos hincapié en la reducción del tiempo de conexión, manteniendo la historia de la sesión de trabajo de cada usuario.

INVESTIGACIONES PRELIMINARES

MARCO TEORICO

USENET

Usenet no es una red formal, sino que es un número de máquinas que intercambian correo electrónico sobre determinados temas, es decir, es un sistema de conferencia y discusión de alcance global. Esta disponible para una amplia variedad de sistemas, pero la forma principal en que se transporta Usenet es a través de Internet o UUCP.

La pieza básica de Usenet es el grupo de noticias o newsgroup el cual es una colección de mensajes con un tema relacionado.

Usenet es muy similar a los mailing list, pero a diferencia de estos, es una red que no se concentra en un único servidor que distribuye los mensajes sino en una cadena de servidores que 'pasan' los mensajes de los grupos que soporta (normalmente los servidores mantienen una cantidad limitada de grupo de noticias, es raro el servidor que tenga todos los grupos disponibles). Esto hace que si un administrador de un servidor decide no soportar un determinado grupo de noticias, poco podemos hacer si dependemos de este servidor únicamente.

Usenet comparte la terminología de los mailing list, cuando se quiere escribir algo que sea leído por los otros lectores del grupo se debe postear o enviar al grupo. Para leer los mensajes dentro de un grupo de noticias se utiliza un software especial llamado "lector de noticias" o newsreader. Por supuesto los browser de la Word Wide Web, permiten también leer grupo de noticias.

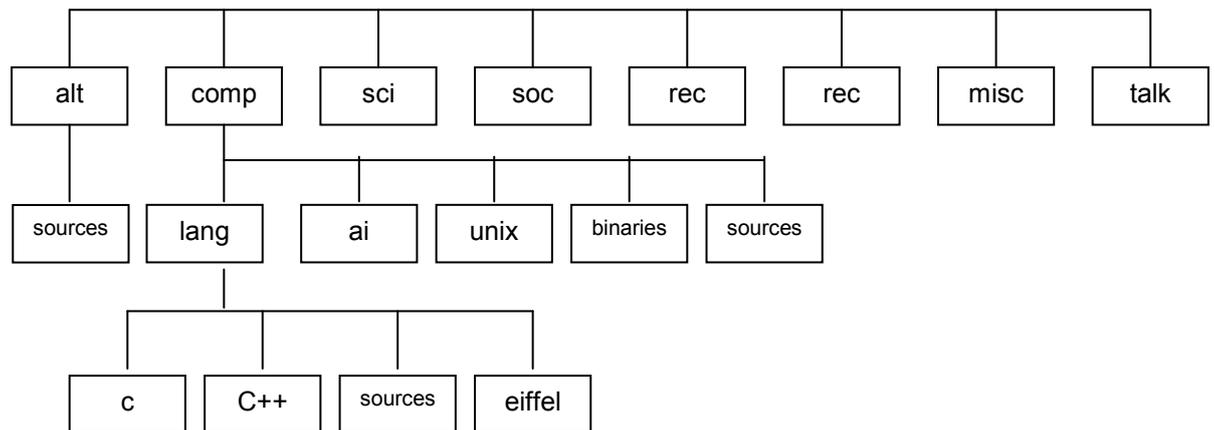
Nadie administra o controla los grupos de Usenet. Una vez creado un grupo de noticias, se puede enviar cualquier artículo al mismo y cualquier usuario podrá leerlo, salvo que el grupo sea moderado, en cuyo caso nuestros artículos pasaran por la censura del moderador quien tiene el poder de no admitir el artículo si el tema es irrelevante al grupo o no es de utilidad.

ORGANIZACION DE USENET

Los artículos posteados en la red están organizados por temas dentro de los grupo de noticias. Se ha implementado una jerarquía de temas de diversos niveles. El primer nivel es genérico y

permite establecer temas a los que esta dedicado un grupo, estos a su vez se dividen en varios subtemas, los cuales son vueltos a dividirse y así sucesivamente. Usenet está organizado como un árbol, es como un tronco con varios miembros fijos (los niveles genéricos) los cuales se dividen en brazos, que se vuelven a dividir en brazos más pequeños y finalmente en ramillas.

Gráficamente lo podemos ver de la siguiente manera:



Habiendo miles de grupos de discusión, es imprescindible utilizar una convención en sus nombres. Los usuarios de Usenet tratan de ser explícitos en los nombres de los grupos para evitar posts que no corresponden al tema tratado en los mismos. Por ejemplo los newsgroups que comienzan con "comp" tienen temas relacionados a las computadoras. Las principales jerarquías son:

- ⇒ Sci: temas científicos y técnicos
- ⇒ Comp: computación
- ⇒ Soc: temas sociales y discusiones políticas
- ⇒ Misc: grupo de temas y debates diversos
- ⇒ Biz: servicios comerciales y publicidad
- ⇒ Rec: hobbies, juegos y recreación
- ⇒ Bionet: investigación biología
- ⇒ Alt: temas inusuales que no son llevados por todos los sitios.
- ⇒ News: administración de noticias de USENET.
- ⇒ Talk: grupo de debate.

Además algunos servidores de noticias crean grupos de noticias locales a una ciudad, estado o región. Estos grupos suelen circular en forma global, salvo los que tienen nombres que comienzan con alt (alternativos) que tienen una distribución menor y tienden a ser menos formales que otros grupos.

Luego del identificador de grupo sigue el de área principal. Por ejemplo "comp.unix" son grupos limitados a la discusión sobre UNIX. Por otra parte, soc.culture suele agrupar los grupos de discusión culturales sobre todo referidos a países. Normalmente, en el nivel tres se 'detiene' la denominación de los newsgroups, aunque en algunos casos como los grupos de computación (comp) y ciencias (sci) las descripciones suelen ser bastante más extensas, por ej. comp.lang.basic.visual.announce (anuncios y novedades sobre Visual Basic).

Los nombres son muy similares en los grupos alt, como por ejemplo, existe soc.culture.argentina y alt.culture.argentina. Se discuten los mismos temas, pero la versión alt del newsgroup no se distribuye tan ampliamente como la soc. Por otra parte es probable que los grupos alt no tengan su equivalente en los grupos principales, mayormente porque están dedicados a temas locales a los servidores que lo soportan.

TRABAJAR CON LA COMUNIDAD USENET

Hay un conjunto de pautas, denominadas 'netiquette' que nos indican la mejor forma de comunicarnos con los demás dentro de USENET.

COMO COMUNICARSE

⇒ *Nunca olvidar que el otro es humano*: como la interacción a través de la red es por computadoras es fácil olvidarse que del otro lado hay personas. Hay que recordar que gente de todo el mundo esta leyendo nuestras palabras. Gritar o maldecir solo hace que los otros piensen mal de nosotros y no nos quieran ayudar cuando necesitemos de ellos.

⇒ *No culpar al administrador del sistema por el comportamiento de sus usuarios*: a veces se puede encontrar necesario escribir al Administrador del sistema sobre algo que esta relacionado con su sitio. Quizás es un problema del software que no esta funcionando correctamente, o puede haber un usuario cuyo comportamiento requiera algún comentario.

Pero siempre hay que tener cuidado porque el Administrador puede no tener idea de lo que se esta hablando y no tener nada que ver.

- ⇒ *Nunca presumir que una persona habla por su organización:* mucha gente que deposita artículos en USENET lo hace desde máquinas de su oficina o desde la escuela. Nunca se debe presumir que la persona esta hablando por la organización desde la cual esta posteando su artículo a menos que lo diga explícitamente.
- ⇒ *Tener cuidado con lo que se dice sobre otros:* hay que recordar que así como nosotros leemos noticias lo hacen millones de personas, entre los que podrían estar el jefe, el jefe del amigo, la novia del hermano del mejor amigo, etc..

COMO POSTEAR

- ⇒ *Ser breve:* nunca diga en diez palabras lo que se puede decir en menos. Hay que recordar que la gente no quiere leer los mensajes largos. Y en muchos casos es posible setear el software para que no traiga artículos que superen una determinada cantidad de líneas.
- ⇒ *Los artículos reflejan al autor:* la mayoría de la gente sobre USENET sabrá de nosotros únicamente por nuestras palabras, lo que decimos y como lo decimos. Por eso es recomendable cuidar nuestra imagen, ya que en este caso es la única forma en que nos conocerán. Hay que tomarse un tiempo para asegurarse que el artículo no dice nada que nos puede avergonzar. Hay que minimizar los errores de tipeo/ortografía y asegurar que el artículo es fácil de leer y comprender.
- ⇒ *Preguntas específicas:* preguntas demasiado generales son duras para responder. "Como puedo eliminar un archivo cuyo nombre comienza con el símbolo '-' en Unix?" es mejor que "Como puedo eliminar archivos usando el comando rm de UNIX?". Se debe incluir los detalles de cada número de versión y número de modelo en el cual se va a aplicar. Dar una idea específica de lo que se esta tratando de hacer. Recordar que en general, preguntas vagas usualmente reciben respuestas vagas.

-
- ⇒ *Ser sucinto*: si se está posteando sobre un problema o dificultad, se debe tratar de resumir esto en un pequeño ejemplo. Postear un ejemplo de cientos de líneas con el comentario "Esto no parece funcionar" a nadie le caerá simpático.
 - ⇒ *Postear en el lugar correcto*: ciertos grupos (como comp.unix.questions y news.newusers.answers) son específicamente designados para preguntas de principiantes. Otros grupos (como comp.unix.wizards y comp.unix.internals) no son explícitamente para este tipo de preguntas.
 - ⇒ *Estar seguro de haber hecho un esfuerzo honesto para encontrar la respuesta por si mismo*: las preguntas que son fácilmente respondidas con un rápido golpe de vista en la documentación, no obtienen simpáticas respuestas. Peor aun si la respuesta a la pregunta es una respuesta en el FAQ.
 - ⇒ *Es correcta*: es aconsejable chequear dos veces las respuestas. Si esta envuelve un comando, tratar de probarlo en el propio sistema. Si es una dirección o número de teléfono, chequearlo antes de postearlo.
 - ⇒ *Dar información sobre diferencias entre sistemas*: no todo el mundo usa el mismo equipamiento. Se debe estar seguro que no se está posteando la respuesta correcta para una Mac cuando la pregunta fue sobre PC.
 - ⇒ *Usar títulos descriptivos*: la línea de subject de un artículo le permite a la gente con una limitada cantidad de tiempo decidir leer o no el artículo. Se debe decir a la gente sobre que se trata el artículo antes que lo lea. Un titulo como "auto en venta" en rec.autos no ayudará tanto como "R19 en venta". Sin embargo hay que tener en cuenta que algunos servidores truncan la longitud de la línea subject a 40 caracteres, por lo tanto, hay que mantener los subjects cortos y concisos.
 - ⇒ *Pensar en el auditorio*: cuando se postea un artículo hay que pensar en la gente a la que alcanzará. Preguntar sobre UNIX en rec.autos no llegará a tanta gente como si estuviera en comp.unix.questions or comp.unix.internals. Siempre hay que tratar de encontrar la audiencia mas apropiada y no la mas amplia.
Si el mensaje es de interés solo sobre un área geográfica restringida (departamentos, venta de autos, reuniones, conciertos, etc.) restringir la distribución del mensaje al área local.

Si se quiere tratar de testear algo no usar los newsgroups globales, mensajes en misc.misc del tipo "Esto es una prueba" es posible que ocasione un gran número de mensajes críticos en nuestro buzón. Hay algunos newsgroups que son locales que deberían ser usados en estos casos.

- ⇒ *Postear el mensaje solamente una vez*: evitar postear el mensaje en más de un grupo de noticias, a menos que esté seguro que es apropiado. Si se postea en múltiples grupos de noticias, no postear en cada grupo por separado, sino especificar todos los grupos en una única copia del mensaje. Esto reduce el overhead y permite a la gente que esta suscrito a más de un grupo ver el mensaje solamente una vez.

- ⇒ *Utilizar encriptación de mensajes si es un contenido cuestionable*: ciertos newsgroups (como rec.humor) contienen mensajes que pueden ser ofensivos para algunas personas. Para estar seguro que el mensaje no es leído a menos que sea explícitamente requerido, este debería estar encriptado. El método estándar de encriptación es rotar cada letra por la que está a 13 posiciones delante de ella, esto es conocido como rot13 y cuando se rota un mensaje la palabra 'rot13' debería estar en la línea subject.

- ⇒ *Resumir cuando se esta haciendo un follow up*: cuando se esta respondiendo un artículo para continuar un thread es muy importante incluir parte del mensaje al que estamos respondiendo, para que el lector ocasional (que eventualmente no haya leído el artículo original) pueda saber que estamos respondiendo. La forma mas simple de hacer esto es realizar un resumen del artículo al que respondemos. No incluyamos todo el artículo, pues es un desperdicio de espacio y tiempo de acceso y también molestaremos a los lectores que hayan leído el original. El mismo software se puede encargar de hacer la inclusión del contexto en forma automática.

- ⇒ *Usar mail, no hacer follow up*: como los lectores de USENET son muchos, dada una pregunta es muy probable que dentro del grupo una gran cantidad sea capaz de responderla. Y si todos responden la misma pregunta, el lector ocasional se encontraría con un thread a un artículo con varias respuestas repetidas. Y seguramente a este thread se agregarán mas artículos protestando porque se habrá publicado las respuestas.
Es posible evitar este problema. Lo mejor en el caso de una respuesta a un tema en particular es enviar un mail a quien hizo la consulta, en vez de hacer un posting. De esta forma, el mensaje saldrá solo una vez y el receptor verá satisfecha su curiosidad.

- ⇒ *Resumir las respuestas recibidas*: es muy común obtener muchas buenas respuestas vía mail cuando se ha posteado una pregunta. Si se considera que el tema puede ser de interés general, pensar sobre la posibilidad de resumir las respuestas y postearlas al grupo de noticias donde originalmente se posteo la pregunta. Esto no significa incluir todas las respuestas y headers, sino mas bien, editar e incluir la información de las respuesta a la pregunta y postear un artículo agradeciendo estas y enviando las mismas. Cuando se postea un mensaje resumen, usar la misma línea de subject con la que se ha posteado la pregunta agregando "SUMMARY" al final.
- Si se tiene planeado hacer un sumario de las respuesta que se reciben, mencionarlo cuando se postea. Algunas personas están más dispuestas a ayudar cuando su esfuerzo ayuda a más de una persona.
- ⇒ *Leer todos los follows up y no repetir lo que ya se ha dicho*: antes de hacer un follow up, leer el resto de los mensajes para asegurarse que nadie a dicho ya lo que se va decir.
- ⇒ *Verificar los headers cuando se esta haciendo follow up*: algunos programas lectores de noticias permiten especificar que los follow-ups a un artículo debe ir a un conjunto específico de grupos de noticias posiblemente diferentes al cual originalmente el artículo fue posteado. Si se cambian los grupos nombrados en el header, o si se dirige el follow-up a un grupo particular, hay que tener en cuenta que no todo el mundo lee los headers.
- ⇒ *Proveer referencias sobre otras fuentes de información*: si se usan hechos para soportar una causa, establecer de donde vienen y no usarlas como propias. La documentación técnica no siempre es un modelo perfectamente claro y algunas veces esto no es obvio cuando se mira. Una referencia o dos pueden ser grandes ayudas.
- ⇒ *Limitar la longitud de la línea y no usar caracteres de control*: tratar de mantener el texto en formato genérico. Mucha gente lee Usenet desde terminales de 80 columnas, entonces tratar de mantener las líneas a una longitud menos de 80 para obtener legibilidad. También hay que tener en cuenta que hay muchas formas de terminales en uso. Si se utiliza algún carácter de control especial en el mensaje puede resultar que sea ilegible en alguna terminal, por ejemplo un carácter que hace vídeo inverso en una terminal, puede causar un bloqueo de teclado y modo gráfico en alguna otra. También debería ser evitado el uso de Tabs, ya que podría ser interpretado diferentemente.

⇒ *Derechos de autor, copyrights o como evitar problemas*: no hay que publicar artículos con elementos protegidos por las leyes de copyright. Esto incluye textos, gráficos, música y muchos otros formatos de archivos electrónicos. Algunos lectores de news adoptan la postura ¿y a mi que? ¿ que me pueden hacer? . Esta actitud es muy peligrosa, ya que los lectores pueden sufrir castigos por violar las leyes de copyright. Hay que ser cuidadoso con el material que se postea, si no es de nuestra propiedad. Esta misma advertencia corresponde aplicar al material que se recupera de Usenet. Que alguien haya publicado un material, no significa que este libre de copyright. Ha habido casos resonantes de violación de Copyright debido a individuos que han escaneado fotografías y posteado los archivos correspondientes sin permiso expreso de los autores.

⇒ *Gritos y susurros*: USENET nació en un entorno de terminales en modo texto, así se fueron generando mecanismos para enfatizar lo que se deseaba expresar. Por ejemplo, cuando se desea enfatizar una palabra, pero sin demasiado "énfasis", simplemente se incluía el carácter "_" o bien , los asteriscos "*" ". Ahora bien, si realmente se desea gritar algo, se utilizan las letras mayúsculas. El truco de las mayúsculas es usarlas con moderación, poniendo énfasis en lo realmente importante. Escribir frases largas en mayúsculas se considera de mala educación. Es probable que resultemos víctimas de algún flame si hacemos esto. Si se quiere evitar insultar directamente, se pueden usar caracteres no alfabéticos, al azar, como en los dibujos animados.

⇒ *Smiley (caritas)*: Los smileys o emoticons son esos pequeños símbolos que se suelen agregar al final de un mensaje que deben ser vistos de costado apoyando la cabeza en el hombro izquierdo. El smiley mas famoso es :)

La utilización de smileys en los post debe hacerse con moderación. Si se exagera, nuestro artículo terminara pareciendo tonto. Sin embargo un smiley puede ser muy útil para remarcar un sarcasmo o una frase ingeniosa, o modificar el tono de una frase como "eso fue realmente tonto :) donde queremos indicar que la frase fue tonta pero lo decimos con buena onda no agresivamente.

Algunos smileys populares son

:-) :) feliz

:-(triste

:-< , :-c muy triste o deprimido

:-o admiración o sorpresa

:-\ duda o desconfianza
:-@ gritando
:-} carita de santo
;-) guiño

⇒ *Luego de haber leído un grupo por un tiempo, comenzar a postear:* se debería estar familiarizado con el grupo antes de postear un artículo en él. No se debe postear en grupos de noticias que no han sido leídos o no se han leído unos pocos artículos, ya que no se sabrá las convenciones y temas del grupo. Uno normalmente no se junta a una conversación solo porque va caminando, sino que primero escucha y solo habla si tiene algo apropiado que decir.

QUE NO POSTEAR

- ⇒ *Divulgar tarde noticias sobre eventos:* al enterarse sobre la realización de un evento, no se debe correr a la Terminal a postear un artículo diciendo "Has escuchado sobre...?" porque la respuesta es si. No se debe enviar cosas que ya alguien ha escuchado.
- ⇒ *Frequently Asked Questions:* existen un número de preguntas que son frecuentemente solicitadas y ya han sido colgadas de la lista de FAQ. Antes de postear una pregunta a un newsgroup, se debe chequear y ver si el grupo tiene una lista de FAQ y ya existe la respuesta a la pregunta.
- ⇒ *Preguntas sin sentido:* no se deben hacer preguntas las cuales no tienen respuestas, postear un mensaje preguntando, por ejemplo, si es mejor BORLAND o MICROSOFT no tiene sentido y no aumentará el nivel cultural.
- ⇒ *Evitar preguntas generales:* por ejemplo, mi próximo programa lo debería hacer en C++ o Delphi?, Debería comprarme un Ford o un Renault?. Una pregunta que paso a la historia fue la de un usuario que preguntó "Does anyone know what time is it?"
- ⇒ *Mail privado:* generalmente es de mal gusto postear un mensaje privado sin el permiso del que lo envió. Algunas veces el dueño del mensaje explícitamente indica que se puede pasar la información.

DONDE POSTEAR

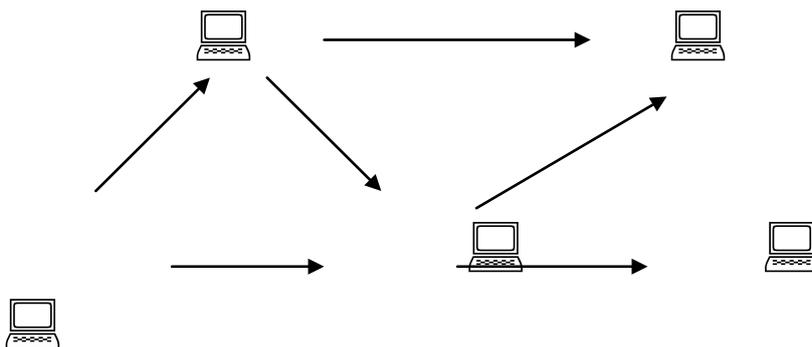
- ⇒ *En grupos de discusión:* los grupos de noticias Source y Binary son para postear fuentes y archivos binarios –no discusiones sobre fuentes y archivos binarios. La mayoría de estos grupos tienen grupos de discusión, usualmente el nombre del grupo agregando la letra d. Por ejemplo, cualquier discusión relacionada con postear en comp.sources es comp.sources.d.
- ⇒ *En grupos advocacy:* estos grupos como por ejemplo, comp.os.ms.windows.advocacy aparecieron con la intención de discutir temas como “Are PCs better than Macs?”. Aún en este caso, es mejor no hacer preguntas con discusiones sin sentido.
- ⇒ *En grupos locales:* tratar que las discusiones locales se cuestionen en grupos locales. No solo se reduce el clutter de la red, sino que se obtendrán mejores respuestas.
- ⇒ *En grupos correctos:* finalmente, tener cuidado de postear en grupos correctos.

DISTRIBUCION DE ARTÍCULOS EN USENET

Resulta interesante en este punto conocer un poco sobre el mecanismo por el cual los artículos son posteados y transferidos entre servidores.

Cuando se postea un artículo a un grupo de noticias, este es almacenado en un archivo en el servidor. Adicionalmente el artículo puede ser transferido a otros servidores de USENET en el mundo, usualmente a las pocas horas, usando NNTP. Cualquier usuario accediendo al servidor tendrá acceso a este artículo.

Además el sitio puede estar de acuerdo con uno o más servidores y cooperar en el intercambio de artículos. Esto significa que dos servidores pueden hacer periódicamente conexiones e intercambiar cualquier mensaje que haya sido posteoado en un servidor pero no en otro.



Como ejemplo consideremos la hipotética red anterior. Cuando un artículo es posteado a un servidor, este es transferido inmediatamente a los vecinos de la red. Estos servidores pasan a su vez el artículo a sus vecinos hasta que el mensaje se haya propagado en todos los servidores de la red. Notar que, un servidor puede recibir los artículos desde varios servidores, en estos casos, se acepta el primer mensaje que llega y se rechaza el del otro servidor.

CONSIDERACIONES SOBRE GRUPOS DE NOTICIAS

Cuando se envía un artículo a un grupo de noticias se tiene un objetivo: distribuir información. El resto de las personas suelen tener sus propios objetivos: perder el mínimo tiempo posible al recuperar los artículos. La calidad de nuestros artículos y su contenido son importantes. Siguiendo algunas recomendaciones es posible llegar a una considerable audiencia en forma apropiada.

FLAMES

Son respuestas insultantes que surgen de algo dicho por alguien en Internet. Algunos flames son realmente intolerables. Sin embargo, no todas las respuestas agresivas son flames. Algunas personas suelen indicar con <flame on> que lo que sigue es un contenido agresivo y emocional. Sin embargo, las respuestas emocionales no son reprobables, solo los ataques a las personas en vez de las ideas son lamentables. Estas guerras de flames suelen ser muy desagradables, aunque a veces pueden llegar a convertir un mal entendido en una nueva concepción de grupo.

Algunas personas se divierten generando guerras de flames. Cuando el ambiente de un thread esta caldeado, responden en forma insultante para generar mas ruido. Un 'guerrero de flames' respondió alguna vez: "... Me gustan las buenas guerras de flames. Internet esta pensado para esto. Si a alguien le disgustan las guerras de flames, simplemente que no lea news..." Por suerte esta clase de individuos es cada vez mas escasa, a medida que la audiencia de los news es mas amplia y variada.

FLAMES Y FAQ

Mas de una guerra de flames se comienza cuando publicamos un artículo consultando sobre algo en un grupo, que ya ha sido respondido. Por eso existen archivos llamados FAQ(Frequently Asked Questions). Estos archivos contienen lo que su titulo indica: las preguntas que realizan casi siempre los principiantes que, se unen al grupo. Los veteranos están cansados de contestar siempre las mismas preguntas entonces colocan las respuestas en FAQ y pretenden olvidarse de ella. Por eso si cometemos el error de publicar un artículo con una pregunta que se encuentra en el FAQ no nos deberá asombrar si somos víctimas de algún flame.

SURFEANDO Y ESPIANDO

En Usenet existen muchos términos que se encuentran frecuentemente, entre ellos 'surfing' y 'lurking'. Los lurkers (mirones) son usuarios que se suscriben a los grupos de noticias, leen sus artículos, pero no hacen post. No hay nada malo en esa actitud, pero los veteranos de Usenet dicen que se pierden lo mejor de los grupos, es decir, participar e interactuar con los integrantes. Esto no quiere decir que debemos responder a cada artículo o generar post siempre, pero no hay que inhibirse cuando sintamos que tenemos algo que decir.

Espiar es útil para sentir la onda de un grupo de noticias, ver como se desarrollan los debates antes de tirarse a la piletta y opinar. Los veteranos nos recomiendan hacer de mirón por lo menos 6 meses antes de comenzar a postear, para tomarle el pulso y el estilo al grupo.

Los surfers, en cambio son lurkers que además recorren varios grupos a la vez. Tampoco hay nada de malo en eso y el termino se aplica ahora mas a los que recorren la WWW pero surgió de Usenet.

COMPORTAMIENTO DE LOS GRUPOS DE NOTICIAS

No existen reglas formales sobre lo que podemos escribir, pero los mismos usuarios y los administradores de los sistemas marcan límites, aunque en la mayoría de los casos nunca puedan imponerse, ellos decidirán que se discutirá y que no.

Supongamos que por alguna razón polemizamos con todo un grupo de noticias. En algunos casos, el administrador del servidor donde dejamos nuestros post (siempre hay un punto de entrada) podría hacerse eco de las quejas y eventualmente cerrarnos la cuenta. Pero si el administrador ignora las quejas, o bien, nuestro servidor no tiene un administrador dedicado a los grupos de noticias, el grupo en cuestión no tiene mas remedio que soportarnos. Pero de la misma manera, los usuarios del grupo de noticias pueden dirigir a nosotros un 'flame war'(guerra de insultos) que puede llegar a saturar nuestra casilla de correo electrónico, así es como el sistema se autoregula.

CREACION DE UN GRUPO DE NOTICIAS

A continuación se detallan las pautas que han sido consideradas como apropiadas en la creación de nuevos grupo de noticias de la jerarquía estándar de Usenet. No intentan crear una política de Usenet, para crear otros grupos en 'alt' o jerarquías de noticias locales no son necesarias, las jerarquías afectadas son misc, comp, humanities, rec, soc, talk, las cuales son las mas ampliamente distribuidas.

Se deben seguir los siguientes pasos:

LA DISCUSIÓN

Se solicita un pedido de discusión en los grupos news.announce.newgroups, news.groups y también se puede mandar a otros grupos o mailing lists relacionados con el tema propuesto.

El nombre y las características del grupo propuesto y si será moderado o sin moderar (y en caso de moderado quien será el moderador) se determinan al tiempo de la discusión. Si no se llega a un acuerdo en este punto entre los proponentes del nuevo grupo, dentro de los 30 dias de discusión, se dará por terminada la misma y los proponentes deberán arreglar detalles afuera entre ellos, preparar una nueva propuesta y volver a comenzar.

Si los impulsores del nuevo grupo necesitan ayuda dentro del procedimiento de creación, existe un grupo voluntario de veteranos experimentados en la creación de grupo de noticias. Ellos ayudan a la gente que quiere proponer nuevos grupos con la creación de una buena propuesta, entonces, se puede enviar un mensaje a groupadvice@uunet.uu.net o mentors@acpub.duke.edu, algunos administradores de newsgroups veteranos ayudarán.

LA VOTACIÓN

El Usenet Volunteer Votetakers (UVV Tomadores de Votos Voluntarios) es un grupo de recolectores imparciales, que se encargan de la recolección y el recuento de los votos para todas las propuestas de grupo de noticias. Existe un responsable encargado de coordinar este grupo, a quien se debe contactar para iniciar el mecanismo de votación, en donde los participantes de la votación decidirán por SI o NO si es aprobado por la comunidad de Usenet el nuevo grupo de noticias.

Después del período de discusión, si se ha determinado que el nuevo grupo es realmente deseado, el nombre y las características son aceptados, se ha determinado si será moderado o no y quien será el moderador, la llamada a los votos será a través de un artículo en news.announce.newgroups y cualquier otro grupo o mailing list en donde pueda haber sido colocado el requerimiento original para la discusión. La llamada para los votos debería incluir instrucciones claras de como lanzar un voto. Debe ser claramente explicado y tan fácil de hacer, tanto el lanzamiento de un voto para la creación, como para el rechazo. Es permitido establecer dos direcciones separadas para los mail por SI o NO, siempre que ambas direcciones sean sobre la misma máquina.

El período de votación no deberá ser, ni menor a 21 días ni mayor a 31 días. Solo se tendrán en cuenta los votos enviados a las direcciones especificadas, los votos colocados en la red por cualquier razón (incluyendo la imposibilidad de comunicarse), no serán tenidos en cuenta.

Los votos no pueden ser transferidos a otras propuestas similares. Un voto contará solamente para la propuesta exacta al que es respuesta. En particular un voto no contará para o en contra de un newsgroups con un nombre diferente.

Los votos deben ser explícitos, deberían ser del tipo "...Yo voto por el grupo ..." o "...Yo voto en contra del grupo ...". En particular respuestas del tipo "...yo podría votar por este grupo si..." solo deberían ser consideradas como comentarios y no contadas como votos.

EL RESULTADO

Una vez, realizada la votación, los resultados se publican en news.announce.newgroups, incluyendo los nombres y direcciones de los votantes (si los hubiera). Luego de publicados los resultados, se deberá esperar 5 días donde se podrán corregir la lista de votantes y/o el procedimiento de votación. Si luego del período de espera no hay objeciones que pudieran invalidar el voto, y si los votos por la afirmativa superan en 100 a los de la negativa, y al menos 2/3 del total de votos recibidos son en favor de la creación, se enviará un mensaje de newgroup a la red. Si no se supera ese margen el newsgroup no deberá ser creado.

El mensaje de grupo nuevo se enviará al moderador de news.announce.newgroups para su publicación definitiva. Si, además el grupo es moderado, se enviará un mail al encargado de administrar los grupos moderados, con el nombre y la dirección del moderador.

En caso de votación negativa, se deberá esperar 6 meses para reintentar la votación. Si el nuevo grupo no pasó la discusión, se podrá reintentar de inmediato

FORMATO DE LOS ARTÍCULOS

Un artículo está formado por tres componentes:

1. El encabezado (header)

Contiene toda la información administrativa necesaria para procesar el artículo y es generado por un programa lector de noticias. Algunos headers, como el subject, consiste en un texto el cual se entra en el momento de hacer la publicación del artículo, otros headers, como el identificador del mensaje, son agregados por el servidor de noticias.

2. El cuerpo

Como una carta, el cuerpo es la parte principal del artículo y consiste en el texto que se escribe.

3. La firma

A pesar de no ser requerido, casi todos los mensajes finalizan con una firma. Usualmente esta no es escrita por el usuario sino que es agregada al final del mismo por el software para publicar artículos. El texto del mensaje es usualmente almacenado en el directorio home del usuario en un archivo llamado .signature. El formato varía pero generalmente tiene el nombre del usuario, la dirección de mail, la compañía o proyecto, título del trabajo y el número de teléfono o fax.

La figura siguiente muestra un ejemplo de un artículo típico en la red de noticias.

```

Newsgroup: news.software.nn
Path: jho!uupsi2!uunet!darwin.sura.net!context!convex!shaver
From: shaver@convex.com (Dave Shaver)
Subject: Re: How can I make nn to auto-unsubscribe?
Sender: USENET@news.eng.convex.com (news access account)
Message-ID: <shaver.714180556@convex.convex.com>
Date: Tue, 18 Aug 1999 23:29:15 TM
References: thut.714173899@vipunen.hut.fi
Nntp-Posting-Host: zeppelin.convex.com
Organization: CONVEX Computer Corp, MPP System Software, Richardson, TX
X-Disclaimer: This message was written by a user at CONVEX Computer
                Corp. The opinions expressed are those of the user and
                Not necessarily those of CONVEX.
Lines: 10

```

Header

```

Tuomo Huttenen writes:
>How do I customize nn, so that it automatically unsubscribes to
>such new newsgroups which have a certain string at the beginning
>of their name?

See the !:N command in the "GROUP PRESENTATION SEQUENCE" of

```

Body

```

/\ Dave Shaver
\\ CONVEX Computer Corporation, Richardson, TX
\/ Internet: shaver@convex.com UUCP: uunet!convex!shaver

```

Signature

Cuando el servidor de noticias recibe un artículo, lo archiva en una base de datos de modo de dar acceso a los programas lectores de noticias y posteriormente pasa el mismo a otros servidores.

Las acciones que realiza el servidor están gobernadas por archivos de control que indican que grupos de noticias son pasados a que otros servidores. Un artículo es pasado a otro servidor si el artículo esta siendo posteado a un grupo de noticias que es enviado a otro servidor, si el artículo tiene una distribución que es enviada a otro host si el nombre no aparece en el header Path del artículo, con esta regla se evita que los artículos estén siendo depositados en forma cíclica.

FORMATO GLOBAL DE UN ARTICULO

Existen varios documentos que definen el formato de un artículo. El primer documento, ahora obsoleto, fue la RFC 850. Mas tarde pequeñas revisiones, produjeron la RFC 1036.

El formato de los artículos de noticias, es una versión mejorada del formato de los mensajes de mail, el cual esta definido en la RFC 822. La intención fue que este sea compatible con el formato del mail, de modo que cualquier noticia válida, sea también un mail válido, pero al formato de artículos se le agregaron otras restricciones con la finalidad de hacerlos más fáciles de procesar.

El formato global de un artículo de noticias consiste de algunos headers, seguido por una línea en blanco y luego el cuerpo del artículo. La línea en blanco, marca el final de los headers y los separa del cuerpo. Los headers deben seguir ciertas reglas especificadas, mientras que el cuerpo tiene solo unas pocas restricciones. Ambos headers y cuerpo son líneas de texto ASCII.

Dentro de la anatomía de un artículo podemos distinguir las siguientes consideraciones de formato:

- ⇒ *Final de línea:* la marca de final de línea depende del contexto, es decir, del tipo de servidor donde el artículo esta siendo almacenado y el protocolo usado para transmitirlo. Sobre servidores UNIX, cada línea finaliza con un carácter nueva línea LF linefeed (carácter decimal 10). La misma convención es usada en artículos enviados vía UUCP. Sin embargo los protocolos internet, incluyendo NNTP, normalmente marcan el final de la línea con una combinación de retorno de carro y nueva línea (CR, código 13 seguido por LF, código 10)
 - ⇒ *Caracteres:* técnicamente una línea puede contener cualquier carácter ASCII excepto NUL(0), LF(10) y CR(13)
 - ⇒ *Subredes y características de lenguaje:* las reglas pueden ser de alguna manera olvidadas dentro de pequeñas subredes, esto es simplemente un conjunto de servidores que intercambian noticias, y están de acuerdo en hacer cosas diferentes entre ellos. En particular, en países donde el idioma no es Inglés, es común que subredes cooperativas intercambien noticias usando códigos carácter non-ASCII para el lenguaje local. Las convenciones para esto varían ampliamente pero los artículos que utilizan caracteres no ASCII, deberían usar MIME (Multipurpose Internet Mail Extension).
- Otro punto con respecto a lenguajes distintos a Inglés, es que varias constantes usadas en los headers, tales como el nombre del mes en el header DATE, son derivados de palabras

inglesas. Estas constantes, no cambian de país a país, las formas derivadas de inglés, son siempre usadas en los artículos en si mismos. El programa Lector de Noticias debe traducir esos strings para los humanos o cuando los recibe de los humanos.

FORMATO DEL HEADER

Todo artículo de noticias comienza con un conjunto de headers que lo describe y contiene información de control para ser usada por los programas lectores de noticias. La primer línea en blanco finaliza los headers, separándolos del cuerpo del artículo. El orden de los mismos no es significativo.

Cada header comienza al inicio de una línea, y muchos de ellos pueden continuar en varias líneas subsecuentes comenzando cada una con un espacio en blanco.

Existe una estructura predefinida para los headers, cada uno comienza con un nombre de header, seguido por dos puntos (:) y algún espacio en blanco (este último no es requerido en mail pero es obligatorio en news) y a continuación se especifica el contenido del header.

Los nombres de los headers no deben contener otra cosa que letras, dígitos y posiblemente guiones como separadores. Los nombres no son sensibles a mayúsculas o minúsculas.

Los lectores de noticias ignoran cualquier header que no reconocen, esto hace posible extender el formato de news sin causar problemas para viejos programas lectores de noticias. Los nombres de headers que comienzan con "X-" son por convención reservados para extensiones no estándares que tienen significado solo para emisores y receptores específicos.

HEADERS OBLIGATORIOS

Un artículo debe contener uno y solo uno de cada uno de los siguientes headers: Date, From, Message-ID, Subject, Newsgroups y Path.

⇒ *Date:* Tue, 18 Aug 1999 23:29:15 -0400

Este header contiene la fecha y la hora en que el artículo fue publicado en un formato específico.

El día de la semana debe ir seguido por una coma, el nombre del día de la semana y el mes, van siempre exactamente con tres letras, el año puede ser dos o cuatro dígitos y si se

especifican solo dos dígitos son siempre interpretados como que comienzas con "19" y los segundos posteriores al símbolo (:) son opcionales.

El -0400 especifica una zona horaria, en este caso uno que esta 4 horas al oeste del Greenwich Mean Time (GMT).

⇒ *From:* shaver@convex.com (Dave Shaver)

El header From contiene la dirección electrónica y posiblemente el nombre completo del autor del artículo. La dirección puede estar primero seguida por el nombre completo encerrado entre paréntesis, o el nombre completo puede estar primero seguido por la dirección encerrada entre "<".

⇒ *Message-ID:* <shaver.714180556@convex.convex.com>

Este header contiene el identificador del mensaje, el cual es un identificador único que distingue el artículo de todo otro artículo de noticias, generalmente es una combinación del nombre del host, el nombre del usuario, la fecha y un número secuencial. Los símbolos < y > son parte del identificador del mensaje.

Artículos con el mismo identificador son tratados como idénticas copias del mismo artículo, aun si no son idénticos.

⇒ *Subject:* Re: How can I make nn to auto-unsubscribe?

El contenido del header "Subject" es una pequeña frase que describe el tema del artículo. El subject por otro lado depende del autor.

Si el artículo es una respuesta de otro anterior, normalmente comienza con "Re:".

Ocasionalmente se podrá encontrar subjects como "RE: RE: RE: How can I make nn to auto-unsubscribe?" pero esta forma no es correcta; los programas lectores de noticias deberían insertar "RE:" solamente si no existe alguno en el subject.

⇒ *Newsgroups:* news.software.nn

El header Newsgroups especifica al grupo o grupos en los cuales el artículo fue postado. El header contiene una lista separada por comas de los grupos sin blancos después de las comas.

Los nombres de los grupos pueden estar en cualquier orden. Ya que la sintaxis no permite espacios en blanco dentro del contenido del header, es imposible que el header Newsgroups ocupe varias líneas.

Nombrar varios grupos en el header Newsgroups es mejor que publicar artículos idénticos en cada grupo. Los programas lectores de noticias pueden detectar artículos que están publicados en más de un grupo y mostrarlo al usuario solo una vez. Sin embargo colocar muchos grupos en el header no es apropiado.

Es posible distribuir un artículo a un grupo que no exista en el servidor del autor, siempre y cuando al menos uno de los grupos que figuran en el header exista en el servidor. Esto se origina debido a que no todos los servidores de noticias contienen todos los grupos, pero a veces es deseable para una respuesta ir al mismo grupo que el artículo original

⇒ *Path:* jho!uupsi2!uunet!darwin.sura.net!context!convex!shaver

Este header indica los servidores de noticias que el artículo ya ha visitado, de modo tal que la transmisión de artículos no sea redundante. Contiene una lista de nombres de servidores, separados por el símbolo(!). El último nombre es el correspondiente al servidor local, este nombre es generalmente la parte que está después de la @ de la dirección de mail del autor.

El servidor de noticias agrega su nombre y un delimitador al contenido del header Path de todo artículo que procesa. Un servidor no pasa un artículo a un servidor vecino cuyo nombre aparece ya en el header.

Un servidor y sus servidores vecinos tienen que ponerse de acuerdo sobre los nombres que utilizarán para el header Path, y cualquier cambio en el nombre debe ser negociado cuidadosamente. El nombre del servidor debe ser único entre todos los servidores que compartirán artículos procesados por este.

HEADERS OPCIONALES

⇒ *Expires:*

Este header especifica cuando un artículo debería ser borrado del servidor de noticias. La sintaxis es la misma del header path.

Si este header no existe la fecha es determinada por los administradores de cada servidor de noticias en los cuales el artículo fue publicado. El administrador del servidor de noticias puede también determinar hasta que punto esta fecha es tenida en cuenta.

Este header tiene dos principales aplicaciones:

1) Eliminar artículos que no son útiles después de una fecha específica, tales como anuncios de eventos, los cuales pueden ser eliminados una vez que el día del evento ha pasado.

2) Preservar artículos que se consideran útiles por largo tiempo, tales como información de ayuda para nuevos lectores del grupo.

El uso de este header es a veces abusivo por personas que indican una fecha de expiración demasiado larga para artículos comunes. Los servidores de noticias tienen sus propias políticas de expiración de artículos y estas no deberían ser cambiadas sin una buena razón.

Cuando se utiliza este header para un día específico hay que tener en cuenta que la hora por defecto es 00:00, entonces si un artículo que publica un evento es útil hasta por ejemplo el 6 de febrero hay que indicar una hora posterior al evento o bien hasta el 7 de febrero.

⇒ *Reply-to:*

Normalmente se responde un artículo a la dirección que se encuentra en el header From. El header Reply-to modifica este comportamiento especificando una dirección diferente. Este header es utilizado cuando el autor sabe que la dirección del From es incorrecta.

⇒ *Sender:*

Este header identifica quien publicó el artículo, en el caso que difiera del autor del mismo. En ausencia de este header se asume que quien publica es el autor, este header es útil para el caso de los moderadores de los grupos moderados que publican en nombre de otras personas.

⇒ *References:*

Este header lista los identificadores de los mensajes de los cuales es una respuesta. La lista esta separada por espacios blancos.

Los artículos que son respuestas de otros, están obligados a tener un header References. La RFC 1036 ha permitido que los softwares lectores de noticias utilicen rutinas que acorten el header References, borrando algunos identificadores de mensajes. Esto debería ser evitado. Pero si es absolutamente necesario acortar el header, nunca borrar el primer identificador, ni los últimos tres (para dar la posibilidad de reconstruir el thread original en el caso que se pierda algún artículo) o cualquier identificador mencionado en el cuerpo de la respuesta.

Los newsreader utilizan esta información para agrupar artículos al mostrarlos.

⇒ *Distribution:*

Especifica limites geográficos u organizaciones para limitar la propagación de un artículo. En principio podría contener una lista de distribuciones, aunque esto raramente sucede. Si este header no es especificado la distribución es el mundo entero.

⇒ *Keywords:*

Contiene una o más frases que describen algún aspecto del contenido del artículo. Las mismas se encuentran separadas por coma.

⇒ *Summary:*

Al igual que el Subject, contiene una frase provista por el usuario el cual solo tiene el significado de resumir el contenido del artículo.

⇒ *Approved:*

Contiene una lista de direcciones de correo y posiblemente los nombres completos de las personas quienes aprobaron la publicación del artículo. Todos los artículos en los grupos moderados deben tener este header indicando que el moderador ha aprobado el artículo.

⇒ *Lines:*

Este header indica el número de líneas en el cuerpo del artículo como un entero decimal. Con el uso de codificación MIME esto no es muy usado ya que indica la longitud del cuerpo codificado en vez del contenido real.

⇒ *Xref:*

Este header indica por donde el artículo estuvo viajando. Normalmente este header es regenerado completamente por cada servidor. Contiene el nombre del servidor, seguido por una o mas referencias. Cada referencia es el nombre de un servidor, dos puntos (:) y el número del artículo.

⇒ *Organization:*

Contiene una pequeña frase que identifica la organización a la cual pertenece el autor. Debería contener información geográfica (por ej. Ciudad y país) cuando no surge en forma obvia del nombre de la organización.

⇒ *Supersedes:*

Este header especifica una lista de artículos que han de ser cancelados. En general este campo es usado cuando el artículo es una versión actualizada de un artículo que ha sido cancelado.

⇒ *Also-Control:*

Identifica un artículo tanto como un mensaje de control como un artículo normal. Un artículo con un header *Also-Control* es transmitido como un artículo normal, pero el contenido de este header es procesado como si estuviese en el header *Control*.

⇒ *See-Also:*

Contiene una lista de identificadores de artículos que están relacionados con este pero que no son sus precursores. Este header provee una forma de agrupar artículos relacionados, como partes de un único documento que tiene que particionarse en múltiples artículos debido a su tamaño.

⇒ *X-:*

Este header no es específicamente utilizado por el protocolo de news oficial, pero es reservado para headers propios del usuario.

FORMATO DEL CUERPO

⇒ *Citas:*

Cuando se responde a un artículo, no es apropiado suponer que todos los usuarios entienden el contexto exacto. El orden en el que los artículos llegan a un servidor de noticias depende de los caminos de transmisión, además a veces los artículos son perdidos, por lo tanto los usuarios pueden leer la respuesta a un artículo sin haber visto el original. Para dar al usuario algún contexto, es común citar algo del artículo original. Esto es hecho colocando cada línea con un carácter > (y no otro). Esto suele resultar en múltiples niveles de > cuando la cita en sí, contiene otra cita.

⇒ *Limites de tamaño:*

En principio no debería haber límites sobre el tamaño de un artículo o la longitud de las líneas. En la práctica es mejor mantener por debajo de los 70-75 caracteres y artículos dentro del límite de aprox. 60Kbytes. Líneas muy largas ocasionan problemas en muchos software lectores y pueden ser truncadas.

Es conveniente mantener estos límites particionando artículos largos en una secuencia de otros más cortos. Cuando se hace esto, el subject de cada artículo debería mostrar el

número total de partes, mas la posición de esta parte dentro de la secuencia total. La convención es finalizar el subject con un indicador de parte tal como (Parte 2/3).

FORMATO DE LA FIRMA

Una firma es un pequeña parte de texto que es agregada en forma automática al final del artículo y contiene información tal como identificación del autor, dirección de mail, dirección postal, etc..

La firma esta separada del cuerpo por una línea que contiene dos guiones (ASCII 45) seguido por un blanco(ASCII 32).

MENSAJES DE CONTROL

Estos son artículos que tiene información de control. Cuando el servidor de noticias recibe un mensaje de control, toma una acción en consecuencia. La ejecución de la acción requerida esta sujeta a las restricciones locales, las cuales pueden causar que el servidor de noticias deniegue el requerimiento o lo derive al administrador para su aprobación.

La presencia de headers Control, Also-Control o Supersedes significa que un artículo es un mensaje de control. Un artículo que contiene un header Also-Control o Supersedes es también un artículo común, mientras que un artículo que contiene un header Control es solo un mensaje de control. Un header Control o Also-control contiene un verbo que especifica la acción a ser tomada, posiblemente seguida por algunos parámetros separados por espacios en blanco que indican los detalles. Un header Supersedes es equivalente a un header Also-Control con un verbo cancel implícito.

⇒ *Cancel:*

Un artículo cancel, requiere que uno o mas artículos previos, sean cancelados o removidos del servidor de noticias tan pronto como sea posible. Estos artículos son identificados por sus Ids los cuales son dados como parámetros.

El cuerpo de un mensaje es un comentario indicando de porque la cancelación fue requerida.

Si el artículo ha de ser eliminado ya ha arribado al servidor, el administrador chequea la dirección del header From del artículo ha de ser borrado y el del mensaje de cancel, como acción de autenticación. Si el artículo no ha arribado aún, no se puede tomar ninguna acción, dado que podría no arribar nunca, entonces el administrador registra una entrada como si el artículo hubiera arribado, de modo que cuando llegue sea rechazado como duplicado.

⇒ *Ihave, sendme:*

Ambos comparten la misma sintaxis, cada uno es seguido por un parámetro que nombra un servidor. El cuerpo de un mensaje *ihave* o *sendme* es una lista de identificadores de mensajes, uno por línea. El nombre es el del servidor enviando el mensaje. El mensaje *ihave* dice que el emisor tiene un artículo con el identificador especificado, el cual puede ser de interés para el receptor. El mensaje *sendme* pregunta al receptor por la transmisión de los artículos especificados.

Estos mensajes de control son normalmente punto a punto, enviados exactamente de un servidor a otro. Un servidor anuncia sus nuevos arribos a otro a través de un mensaje *ihave*, y el receptor requiere los artículos faltantes a través de un mensaje *sendme*.

⇒ *Newgroup:*

Este mensaje de control solicita que un nuevo grupo de noticias sea creado. Los parámetros son el nombre del grupo y una palabra clave optional *moderated* o *unmoderated*. Si no hay segundo argumento se asume *unmoderated*. Un mensaje de este tipo tiene que tener un header *Approved*.

El cuerpo del mensaje es un comentario, el cual los softwares deberían ignorar, excepto si el cuerpo contiene un descriptor de la siguiente forma:

```
For your newsgroups file:  
Rec.arts.comics  
Comic books, animation, visual story telling.
```

El descriptor contiene dos líneas. La primera línea debe ser exactamente como se mostró, mientras que la segunda línea es una descripción del grupo siendo creado. Si el cuerpo contiene un descriptor, el administrador debe agregar la descripción al archivo *newsgroups*. Una descripción para un grupo moderado debe finalizar con "(Moderated)", el espacio en blanco delante del primer paréntesis es obligatorio.

El resto del cuerpo debería explicar el propósito del grupo y la decisión de crearlo.

⇒ *Rmgroup*:

Se usa para eliminar un grupo. Este mensaje debe tener un header approved. El cuerpo del mensaje es un comentario, que debería explicar porque el grupo esta siendo eliminado.

MIME

Es un conjunto compatible de extensiones de la RFC 822, documentadas en las RFC 2045-2049 que describen y estructuran los cuerpos de mensajes no textuales, que contengan mensajes multimediales que puedan incluir audio o imágenes. MIME fue diseñado para el correo electrónico, pero puede ser utilizado para news.

HEADERS

⇒ *MIME-Version*:

Usa un número de versión para declarar que un mensaje esta de acuerdo con MIME y permite distinguirlo de aquellos generados por viejos softwares.

⇒ *Content-Type*:

Especifica el tipo de dato que contiene el cuerpo del mensaje incluyendo cuerpos multipartes donde cada parte tiene su propio tipo.

⇒ *Content-Transfer-Encoding*:

Especifica como el cuerpo (o parte del cuerpo) ha sido codificado.

⇒ *Content-ID*:

Asigna un identificador único a las partes del cuerpo, de modo que puedan ser referenciadas por las otras partes del cuerpo.

⇒ *Content-Description*:

Similar al header subject, pero describe una parte específica del cuerpo.

TIPOS DE CONTENIDO

Hay una gran variedad de tipos de contenido MIME. Un contenido MIME es especificado como un tipo y un subtipo separados por (/) en el header Content-Type. Información auxiliar específica del subtipo. Por ejemplo:

Content-Type: text/plain; charset=ISO-8859-1

Esto dice que el cuerpo (o parte) es texto y específicamente es un texto plano en un conjunto de caracteres particular (ISO88591)

La intención del sistema de type/subtype es que el tipo especifica la naturaleza general de los datos mientras que el subtipo da mas detalles.

Los tipos son:

⇒ *Text:*

Información textual.

Los principales subtipos son:

1. Plain: texto común, en un conjunto de caracteres especificados (por default ASCII). Text/plain es el contenido por default.
2. Enriched: texto que soporta control de alíneación y atributos tales como subrayado.
3. Html:

⇒ *Multipart:*

Varias partes posiblemente de diferentes tipos, relacionados en alguna forma específica.

Los principales subtipos son:

1. Mixed: las partes son independientes en algún orden específico.
2. Alternative: las partes son diferentes formas de expresar el mismo contenido, ordenados de peor a mejor, de modo que los lectores con diferentes capacidades puedan usar la mejor versión que su equipo soporte.
3. Digest: igual que los mixed pero compacto.
4. Parallel: partes independientes en ningún orden específico para ser presentadas simultáneamente.

⇒ *Message:*

Un mensaje de mail o news.

Los principales subtipos son

1. rfc822: un mensaje de correo
2. partial: un fragmento de un mensaje mas largo, con identificación y otros detalles que permiten unirlo.
3. external-body: una referencia a un body que puede ser obtenido por algún otro medio, tal como FTP.
4. news : un artículo news, según la RFC1036 o sus sucesores.

⇒ *image:*

Un gráfico.

⇒ *audio:*

Sonido.

⇒ *video:*

Imágenes en movimiento, posiblemente incluyendo audio.

⇒ *application:*

Datos que no caen en ningún otro tipo y que solo son comprensibles en un software específico.

Algunos subtipos son:

1. octect-stream: datos binarios
2. postscript: un programa postScript, típicamente el programa debería generar la impresión de un documento.
3. news-transmission: un artículo de news, para ser retransmitido a un destino.

Subtipos que comienzan con X- no son subtipos standares con un significado específico al emisor y receptor. Un uso común es application/X-program, donde se espera que el receptor tenga el programa al que se hace referencia.

CODIFICACIONES DE TRANSFERENCIA DE CONTENIDO

Mucho de los tipos de contenido MIME requieren la habilidad para transportar datos binarios, lo cual no es posible con mail o news sin alguna clase de codificación. Así el header Content-Transfer-Encoding indica como el contenido ha sido codificado.

⇒ *7bit*:

Sin codificación. El contenido es un conjunto de líneas de texto ASCII, cada una de a lo sumo 1000 caracteres. (es la codificación por default).

⇒ *8bit*:

Sin codificación. El contenido son líneas de texto de menos de 1000 caracteres, conteniendo códigos con el 8 bit seteado.

⇒ *binary*:

Sin codificación. El contenido es una secuencia arbitraria de bytes. Ni mail ni news pueden transmitir esto en forma confiable.

⇒ *quoted-printable*:

Principalmente texto ASCII pero contiene algún carácter no-ASCII o alguna línea demasiado larga. La secuencia =XX, donde XX es un valor hexadecimal, codifica un carácter no ASCII. Un = seguido por un final de línea indica que el final de línea no esta presente en el dato original.

⇒ *base64*:

Cualquier secuencia de bytes codificados representando cada grupo de 3 bytes como 4 caracteres del alfabeto A-Z, a-z, 0-9, +, /

LECTORES DE NOTICIAS

La actual pieza de software usada para leer noticias en USENET es llamada lector de noticias o newsreader. Los lectores de noticias vienen en todas las formas y tamaños, pero todos ellos tienen cosas en común.

Ellos mantienen detalles de los grupos suscriptos y que artículos han sido leídos. Presentan nuevos artículos y permiten leer, salvar y responder a ellos.

Algunos de ellos son simples, con solo suficiente funcionalidad para leer mensajes y responder a los mismos, otros tienen lenguaje de macro con cientos de funciones que permiten personalizar cada aspecto de las operaciones de los lectores de noticias. Algunos son escritos para desarrollos particulares como por ejemplo el X Window System, o estar corriendo desde dentro de otros ambientes, como es Emacs.

Todos los lectores de noticias tienen las mismas características básicas pero varían mucho en sus opciones avanzadas y en su interface.

Para poder obtener los grupos de noticias es necesario indicarle al lector de noticias el servidor de donde se pueden tomar.

Actualmente los usuarios de Usenet interactúan usando un gran número de paquetes de software y programas. La mayoría de los lectores ofrecen una integración de dos importantes funciones utilizadas en redes de computación: programa mail y newsreader.

Existe software para leer grupos de noticias para casi cualquier entorno operativo, Windows, DOS, Amiga, Macintosh.

FUNCIONALIDADES PRINCIPALES

Los distintos lectores de noticias que encontramos hoy en el mercado, cuentan con la mayoría de las siguientes funcionalidades

- ⇒ *Postear artículos*
- ⇒ *Leer artículos*
- ⇒ *Buscar artículos*
- ⇒ *Responder artículos*
- ⇒ *Grabar artículos*

- ⇒ *Filtrar artículos*
- ⇒ *Suscribirse a un grupo de interés*
- ⇒ *Seguir Threads*
- ⇒ *Organizar los news y mensajes mail en carpetas personales*
- ⇒ *Enviar y recibir archivos attachados en varios formatos*
- ⇒ *Procesamiento Multi-tarea*
- ⇒ *Utilización de mail*
- ⇒ *Seteo de lenguaje y corrección ortográfica*
- ⇒ *Depuración de Base de Datos*
- ⇒ *Conexión Online/offline*
- ⇒ Importar y exportar archivos newsrc

CONTROL DE LOS ARTÍCULOS QUE SE LEERAN

- ⇒ *El archivo de inicialización .newsrc*

Este archivo es el método principal para controlar los artículos leídos. Fue creado con los primeros lectores de noticias y actualmente se encuentra utilizado por todos los lectores de noticias existentes en el mercado.

El archivo newsrc mantiene una traza de dos cosas: los newsgroups a los que se esta suscripto y que artículos en el newsgroup fueron leídos.

El formato del archivo .newsrc es simple. Hay una sola línea en el archivo para cada grupo de noticias del servidor. El primer item en la línea es el nombre del grupo, seguido por dos puntos (:) o por un símbolo de exclamación (!). Si el nombre es seguido por dos puntos, se está suscrito al grupo y se verán los artículos en el grupo cuando se lea el news. En cambio si el nombre está seguido de un símbolo de exclamación, no se está suscrito al grupo y no se verán los artículos.

Seguido al nombre del grupo se encuentra una lista de números. Estos son los números de los artículos que han sido leídos. Esta lista es actualizada cada vez que se lee news.

Ejemplo:

```
news.announce.newsusers: 1-4957  
news.admin!
```

```
news.announce.conferences:
news.announce.important: 1-46
news.groups!
news.list:
news.lists.ps-maps!
news.misc:
news.newsusers.questions:1-7105
news.software.nn!
news.sysadmin!
comp.ai:1-5513
comp.ai.neural-nets!
comp.lang.pascal!
comp.lang.c++:1-9531
comp.lang.java:1-7829
```

⇒ *Expiración de los artículos*

El software administrador de news borra o expira los artículos news con la regla "oldest file first" y diferentes grupos pueden ser expirados en diferentes condiciones.

Esto tiene el beneficio adicional de conservar espacio en disco. Cuando no se lee un news por un período de tiempo, el backlog de artículos no leídos puede comenzar a agotarse. Pudiendo deletar estos artículos viejos con una regla regular, puede hacerse mucho más fácil el mantenimiento.

El scheduler de expiración varía mucho, de acuerdo al espacio en disco, a la popularidad del grupo de noticias (los grupos más populares suelen mantenerse mas tiempo) y la naturaleza del grupo (los grupos no serios usualmente son expirados rápidamente). El administrador de news puede proveer detalle en el scheduler de expiración para cada sitio.

⇒ *Archivos Kill*

Complementando el archivo .newsrsrc, muchos newsreaders tienen otro mecanismo para controlar que artículos se verán, llamado file kill. Este archivo es usado para filtrar artículos que sabemos por anticipado que no deseamos ver.

Este filtro se puede hacer por subject, por ejemplo no leer cualquier artículo con el subject "Disco Revival", o por autor, por ejemplo, no leer artículos posteados por Joe Obnoxious". Los archivos kill son también usados para seleccionar automáticamente artículos de interés. Cada programa lector de noticias tiene un diferente método para manejar archivos kill. A continuación presentamos una lista con algunos consejos de como usar el archivo kill para maximizar la eficiencia leyendo news.

- Usar el archivo kill para seleccionar por tópicos y autores.
- No molestarse en killiar artículos de autores anónimos.
- Usar la característica kill a los 30 días.
- No poner frases comunes en el archivo kill pues se puede cometer el error de killiar más de lo que se desea.

NN NEWSREADER

HISTORIA

NN es el popular lector de noticias escrito por Kim Storm, su primera versión fue liberada en Dinamarca en 1984, en Europa en 1988 y en el resto del mundo en 1989.

Nn fue diseñado para minimizar el tiempo requerido en leer un news por la ayuda de ignorar los artículos que no eran interesantes.

CARACTERÍSTICAS

- ⇒ Interfaz full-screen point-and-shoot que permite al usuario navegar rápidamente a través de gran número de artículos. La mayoría de los comandos son entrados con una simple tecla.
- ⇒ Tiempo de respuesta rápido, Nn acelera esta operación por el mantenimiento de una base de datos de todos los artículos online.
- ⇒ Poderoso y fácil uso del sistema de kill file. Esta característica permite automáticamente sacar de la pantalla los artículos de poco interés y automáticamente seleccionar artículos preferidos.
- ⇒ Automatiza el manejo de news digests. Los digests son automáticamente explotados y presentados como artículos ordinarios.
- ⇒ Extensivo help online y manual de referencia.
- ⇒ Incorpora facilidades para descompactar y decodificar artículos.

COMIENZO

Para comenzar simplemente correr el comando `nn`, si es la primera vez que se leen news o si por mucho tiempo no se han leído, es capaz de descartar los miles de mensaje que se han acumulado en el sistema durante la ausencia. Para hacer esto, se comienza `nn` con el comando

```
% nn -anumero
```

donde `numero` es el número de artículos que se desean ver de cada grupo.

MODOS

`Nn` tiene dos modos para leer artículos. El primero, `article-selection`, es el modo usado para seleccionar artículos; el segundo, `reading`, es el modo usado para leer los artículos seleccionados.

`Nn` comienza en modo `article-selection` dentro del sitio del primer grupo que contiene artículos no leídos. Entonces se seleccionan los artículos que se desea leer; después de seleccionar los artículos, `nn` entra a modo `reading`, el cual muestra los artículos uno por vez, permitiendo navegar a través de ellos, salvarlos, etc..

Una vez que se haya finalizado de leer los artículos seleccionados en el grupo, se puede pasar a modo `article-selection` para el próximo grupo.

SELECCIONANDO ARTÍCULOS

Al entrar al grupo de noticias nos encontramos en modo `article-selection`, donde seleccionamos los artículos que deseamos leer. Una típica pantalla de selección es la siguiente:

```

Newsgroup: comp.lang.c                Articles: 463 of 5635/48

a Alan Bowler          20      >>>array element swap help
b Lars Wirzenius      28      >linking gcc code with CC code
c A. Kardanev         58      >
d Wayne Throop        9      >>>Editor mined (pointer declaration snafu)
e Thomas Wolff        75      >>>>
f Tim Pierce           56      >>>
g beshir@vax1         11      HELP!! With wordprocessor font
h Mary Wyllie         59      Looking for ides to tran<>things to function call en C.
```

```

i T S Thompson      19      >good memory allocation strategies?
j John Anderson     9       PC Based C Compiler targeted to Mips R3000
k Deepak            44      Function name : C preprocessor question
l Dave Schaumann    30      >
m Dave Sisson       24      Here's some pointers for TV an TMenuItem
n William Kaufman   39      >>>Pointers to freed memory

```

```
-- 20:01 - SELECT - help:? -----24% -----
```

En la parte superior se muestra los grupo (comp.lang.c) y algunas estadísticas como el número de artículo dejados para leer (463 artículos no leídos en este grupo, 5635 artículos están leídos en todos los grupos y 48 grupos dejados para leer).

En la línea de la parte inferior se muestra la hora, el modo (article-selection), el porcentaje que se esta viendo de la lista de artículos (aproximadamente 24%) y un recordatorio que presionando la tecla "?" se muestra la pantalla de help.

Todas las otras líneas de la pantalla describen nuevos artículos. Cada línea consiste de una letra de selección (en la izquierda), el nombre del autor del artículo, el número de líneas en el artículo y el subject del artículo. Si el subject del artículo comienza con el signo ">", el artículo es una respuesta a un artículo anterior; ">>" respuesta de respuesta y así sucesivamente. Si el campo subject solo contiene signo ">" pero no texto, el artículo tiene el mismo subject que uno anterior. Nn ordena el modo de selección mostrando por thread; esta forma hace que sea más fácil ver donde empiezan y terminan los thread.

Para seleccionar un artículo para leerlo, se debe presionar la letra que lo identifica. Frecuentemente se puede desear seleccionar un grupo consecutivo de artículos, entonces presionar un rango de letras identificatorias separadas por '-'. También es posible seleccionar un artículo y todas sus respuestas, tecleando la letra del artículo seguida de '*'.

Para pasar de pantalla, se debe presionar la barra espaciadora y con ">" se avanza una página hacia adelante y con "<" se retrocede una página. Si nos encontramos en la última pantalla de artículos y presionamos la barra espaciadora, nn automáticamente pasa a modo reading, también presionando la tecla "z" o "x" se entra a este modo inmediatamente.

Tecleando "N" se entra al menú del modo de selección para el próximo grupo suscrito, sin leer los artículos que se tenga seleccionado; asimismo, "P" lo hace para el grupo anterior.

BÚSQUEDAS POR SUBJECTS, AUTORES Y MENSAJES

Nn permite hacer búsquedas en varios niveles article-selection (por el texto del subject o nombre del autor) y mensaje (con el contenido de un particular posting). No importa que tipo de búsqueda se esta haciendo, la misma es case-insensitive.

En el nivel article-selection, cuando se hace el match por uno o más líneas de subject o autores, estos posting son seleccionados automáticamente y pueden verse si se los elige. Por algunas razones, nn trata búsquedas por nombres de autor diferente que por subject.

Cuando se está viendo una lista de mensajes de un grupo en particular, se puede buscar a través de líneas de subject tipeando el símbolo "=" y a continuación el texto a buscar. Al tipear el símbolo "=" aparecerá al final de la pantalla un prompt de búsqueda "Seleccione la expresión".

LEYENDO ARTÍCULOS

Cuando se llega al final del menú de selección, nn entra a modo reading, en el cual se puede ver los artículos seleccionados. A continuación se muestra una pantalla típica en modo reading:

```
Jonathan Kamens: Welcome to news.newusers.questions! (weekly posting) 993 01:00
Archive-name: news-newusers-intro
Version: $Id: new-newusers-intro, v 1.20 1993/06/09 15:43:35 jik Exp $
```

```
Welcome to the news.newusers.questions newsgroup! According to the "List of Active
Newsgroup" posting in news.announce.newusers, the purpose of this newsgroup is "Q & A
for users new to the Usenet." So if you've got questions about the USENET, this is the
place to post them!
```

```
Get to know news.announce.newusers.
```

```
However, before you do that, there is another newsgroup with which yo should become
acquainted. The news.announce.,newusers newsgroup contains (once again according to the
"List of Active Nesgroups" provide a base set of information with which all participants
in the USENET should be familiar in order to make the USENET a better place for all of
us.
```

```
-- 05:03 --.questions-LAST - help:?-Top6%--
```

La línea superior muestra el autor del artículo (Jonathan Kamens), el subject (Welcome...), el número de líneas en el posting (993) y la hora que fue posteado (1:00). En la parte inferior de la pantalla una línea similar a la de la pantalla article-selection. El nombre del corriente grupo de noticias (truncado para salvar espacio) ha reemplazado el indicador del modo selected y un contador de los artículos remanentes es mostrado (en esta pantalla, LAST es mostrado porque no hay más artículos seleccionados para leer).

Presionando la barra espaciadora se pasa a la siguiente pantalla del artículo, si está en la última pantalla, ésta toma el próximo artículo seleccionado y si no se tienen más artículos seleccionados en el grupo, presionando la barra espaciadora se entra a modo article-selection para el próximo grupo. Presionando las teclas "delete" o "backspace" se pasa a la página previa. Tipeando "n" en cualquier momento se pasa al próximo artículo y con "p" al anterior.

Existen muchos más comandos –para salvar artículos, decoding artículos, etc..

NAVEGANDO

Para leer un grupo particular se utiliza el comando "G" tanto en modo selection o reading. Al tipear el comando "G", nn solicita el nombre de un newsgroup:

```
Group or Folder (+./~%=sneN) rec.music.folk
```

Luego nn solicita el número de artículos que se desea leer:

```
Number of articles(uasne) (a) 30  
Use: j)ump u)nread a)ll @)archive s)ubject n)ame e)ither or number
```

Presionando la tecla ENTER acepta el default que es "a" (todos), quiere decir que se desea leer todos los artículos en el grupo. Si se elige esta opción, nn entra al modo selection-article, con un menú incluyendo todos los artículos en el newsgroup. Como puede ser muy grande, se elige el número de artículos para leer (30). Alternativamente se puede elegir "s", en ese caso nn solicita un subject y solo se ven los artículos que machean con él. Con "n" se permite seleccionar artículos posteados por una persona dada. Si ya estabamos suscritos en un grupo, se puede seleccionar "u", el cual muestra todos los artículos no leídos, o "j" para saltar directamente al grupo. Nuevos artículos son mostrados como si hubiésemos entrado normalmente.

Otra característica práctica es que nn puede procesar news y mails del mismo modo que el proceso de newsgroups. En la solicitud Group or Folder se debe entrar un nombre de archivo. Este archivo es entonces abierto y se muestra de la misma forma que un newsgroup, el nombre que se ingrese debe comenzar con "." si el archivo se encuentra en el mismo directorio (./unix) o con "~" si está en el directorio home (~/.mbox) o con "/" para indicar la ruta (/usr/spool/mail/mbarriso).

OBTENER AYUDA

Nn provee varios caminos para obtener ayudas, en ambos modos –select o reading– existen pantallas de ayuda; para que se despliegue la pantalla de ayuda en cualquiera de los modos que se este usando, se debe tipear "?".

Pantalla que se despliega en modo SELECT

SELECT (toggle)		MOVE	
a-z0-9	Specified article	,	Next menu line
x-y	Range x to y	/	Previous menu line
x*	Same subject as x	SPACE	Next menu page
.	Current article	<>	Prev/Next menu page
@ ~	Reverse/Undo all selections	^\$	First/Last menu page
=regexp	Matching subjects (= . Selects all)		
L/JJJJ	Leave/Change atributes		
SHOW SELECTED ARTICLE			
SPACE	Show (only when on last menu page)		
Z	Show Now, and return to this group afterwards		
X	Show NOW, and continue with next group		
GOTO OTHER GROUPS			
X	Update current group, skip to next.	Y	Group overview
N P	Update current group, skip to next	~/./nn/init:	
G	Goto named group or open a folder		Define groups
B A	Go back/forward in groups already read.		Presentation sequence.
MISCELLANEOUS			
U C	(Un)suscribe / Cancel	:man	Online manual
F R M	Follow-up/Reply/Mail	:help	More online help
S O W	Save articles	j	Shell escape
:post	Post new article	"	Change menu layout
:unshar	:decode :path Unpack articles	Q	Quit nn
Hit any key to continue			

Finalmente, se puede tipear ":man" para leer la versión online del manual, el cual es presentado en formato grupo, con cada sección presentada como un artículo.

EMERGENCIAS

Al tipear CTRL-G cancela cualquier operación que este en proceso y retorna a cualquier operación que se estuviera previamente haciendo. Esto es útil si se llegara a confundir en el comando a tipear o accidentalmente se presionara el comando que no se desea, pero no es extremadamente práctico porque muchos comandos son ejecutados rápidamente y en el momento de darnos cuenta de un error a lo mejor ya es demasiado tarde para volver atrás.

Tipeando CTRL-L se refresca la pantalla. Esto es útil si la pantalla está sobre cargada por alguna razón.

Como dijimos anteriormente :q es la ventana de escape de emergencia, se usa esto si se desea salir inmediatamente sin actualizar los estados de los archivos.

DECODIFICANDO ARTÍCULOS

Una de las potencias de nn es la facilidad para desempaquetar y decodificar software posteados en USENET. Los comandos que soportan esto son:

:unshar

Decodifica el código fuente y otros archivos que han sido empaquetados en formato shar.

:decode

Decodifica archivos binarios que han sido codificados.

:patch

Automáticamente aplica parches de software que han sido posteados en formato patch.

Ejemplo de :unshar

Se debe entrar el directorio donde deseamos que se realice el comando unshar. Si el directorio no existe, nn pregunta por la confirmación antes de crearlo. El archivo es codificado en ese directorio y el reporte de estado es mostrado cuando la operación finaliza. Son creados dos archivos: Unshar.headers, el cual contiene todas las partes preliminares del artículo y Unshar.result, el cual tiene un log y resultado final de la operación unshar.

En el modo article-selection existe la flexibilidad de decodificar multipart postings. Se deben seleccionar los artículos que se desea desempaquetar y emitir el comando :unshar. Como en modo reading nn solicita el directorio y si no existe pregunta si se desea crearlo.

Una vez desempaquetados los artículos son deseleccionados.

POSTEANDO ARTÍCULOS

Existen dos caminos para postear artículos en nn. Se puede postear un mensaje original usando el comando :post, o se puede responder el mensaje posteadado por alguien con el comando "f" (follow-up). Estos comandos pueden ser ejecutados en cualquier modo, selection or reading.

⇒ *Posteando nuevos artículos*

Al emitir el comando :post, nn hace algunas preguntas y entonces luego da lugar al editor para que se escriba el texto del mensaje.

```
:post
```

comienza el proceso de posteo

```
POST to group:misc.test
```

Se debe entrar el nombre del grupo al cual se va a postear. Luego presionando la tecla "enter" se selecciona el default, lo cual significa que se va a postear en el grupo corriente

```
Subject: test post
```

Se debe entrar un breve subject en esta línea. Se debe tratar que sea lo más específico posible.

```
Keywords: enter
```

Este texto es puesto en el campo keyword del header del mensaje, el cual puede ser usado para buscar mensajes por tópico, o construir índices en el archivo news.

```
Summary: this is a test
```

Se debe ingresar un pequeño resumen del mensaje.

```
Distribution: (default 'world') enter
```

Este campo controla la distribución geográfica del mensaje.

Una vez que se hayan respondido todas estas preguntas, toma lugar el editor para escribir el texto del artículo.

```
Newsgroups:misc.test
Distribution:world
Subject: test post
Summary: this is a test

This is only a test!
```

Una vez finalizada la escritura del texto del artículo, se graba en un archivo temporario y se sale del editor. En este momento se puede tomar las siguientes acciones:

- Abortar el posting.
- Enviar el artículo por mail en lugar de postearlo
- Editar el artículo
- Ver el artículo como aparecerá
- Guardar el archivo en disco sin postearlo.
- Postear el artículo (default)

```
a)abort e)dit m)ail r)edit s)end v)iew w)rite
Action: (post article) p
```

Dependiendo del servidor, puede tomar varios minutos o todo un día para que el posting comience a propagarse en otros sitios.

Si se decide más tarde que el posting era inapropiado, se puede volver atrás, se debe esperar que el artículo aparezca en el server, se selecciona el mismo y se tipea el comando "c" para cancelarlo.

⇒ *Respondiendo un artículo*

El procedimiento para responder un artículo es esencialmente el mismo que para postear un nuevo artículo. La diferencia está en:

- Typear "F" para responder a un artículo
- Si se está en modo article-selection, nn pregunta a cual artículo se desea responder. Si se está en modo reading, nn asume que se desea responder al artículo que se está leyendo.
- Se da la opción de incluir el artículo original en el artículo.
- Si no se ingresa el nombre de newsgroup, la línea de subject o la keywords, se toman estos datos del artículo al cual se está respondiendo. Aunque una vez que estemos en el editor, se pueden modificar los mismos si se desea.

ENVIANDO UN MAIL

Nn provee algunos servicios básicos para el envío de correo electrónico. Se puede usar nn para originar nuevos mensajes de mail, aunque esto es raramente utilizado, comúnmente se utiliza para responder a un autor de un determinado artículo o para reenviar un artículo interesante a quien no puede leer news.

⇒ *Responder*

La razón más común para utilizar mail con nn es responder al autor de un artículo sin enviar un mensaje a todo el newsgroup.

Para enviar una respuesta, se tipea "R" o se utiliza el comando :reply en cualquiera de los dos modos article-selection o reading.

Si se encuentra en modo selection, nn pregunta a que artículo se desea responder.

```
Reply to author of article:
```

Si se encuentra en modo reading, nn asume que se desea responder al autor del artículo corriente.

Luego se pregunta si se desea incluir el texto del artículo original:

```
Include original article?
```

Si se responde "y", nn incluye el artículo original en el mensaje.

⇒ *Reenviando un artículo vía mail*

Se puede reenviar un artículo interesante a personas que no tienen la posibilidad de leer news o también es práctico utilizar esta opción para guardar artículos.

Para reenviar un artículo se siguen los siguientes pasos:

- Se invoca al comando mail usando "M" o :mail
- Si se está en modo reading se tepea "y" cuando nn pregunta si se desea reenviar el artículo.
Si se está en modo selection, se indica cual es el artículo que se desea reenviar cuando nn pregunta por su nombre.
- Presionar la tecla "enter".
- Presionar "n" cuando se pregunta si se desea editar el mensaje enviado.
- Nn entonces pregunta si se desea agregar la firma al mensaje.

KILLIAR Y SELECCIONAR ARTÍCULOS

Cuando se esta leyendo news, frecuentemente se encuentran artículos aburridos, poco interesantes u incluso ofensivos. Nn realiza fácilmente el auto kill de tales artículos y el auto select de aquellos artículos interesantes. Esto puede hacerse en un grupo de noticias particular o a lo largo de todos los grupos en forma permanente o temporaria.

⇒ *Auto kill o auto select de artículos*

Para auto kill o auto select de artículos, presionar "K" en cualquiera de los modos article-selection o reading. Nn entonces hace un serie de preguntas, en el ejemplo siguiente se elige kill los artículos permanentemente con subject "boring post" en el grupo misc.test. La línea que a continuación se muestra varía de acuerdo a la opción elegida:

```
AUTO (k)ill or (s)elect (CR => Kill subject 30 days)
```

Presionando la tecla return nn kills un subject particular por 30 días.

Si se quiere killiar un artículo permanentemente (k) o tener una selección automática (s) se realizarán más preguntas:

```
AUTO KILL on (s)ubject or (n)ame (s) s
```

Se debe elegir si se va a kill un artículo por subject o por nombre.

```
KILL Subject: (=/) boring post
```

Con la tecla "/" se puede killiar los subject que contengan la expresión o con igual subject con la tecla "="

```
KILL in (g)roup 'misc.test' or in (a)ll groups (g) g
```

Se puede killiar un artículo en el corriente grupo tipeando "g" o en todos los grupos tipeando "a".

```
Lifetime of entry in days (p)ermanent (39) p
```

Se puede decir que el kill sea permanente o por un período de días

```
CONFIRM KILL Subject Permanent: boring post y
```

Se presiona "y" para confirmar o "n" en caso contrario.

⇒

⇒ *Unkilling*

No hay comando para deshacer un artículo killiado o seleccionado. Lo que se puede hacer es ver o modificar la selección del archivo kill, editar el file ~/.nn/kill.

Hay una sola especificación por línea de kill o select. Se debe encontrar el subject o nombre que se desea unkill y borrarlo. Nn lee este archivo cada vez que inicia, así que tomará efecto la próxima vez que corra nn. También puede tomar efecto usando el comando :compile, el cual causa que nn relea el archivo kill.

PERSONALIZAR NN

Nn provee varias características que pueden ser personalizadas según la preferencia como secuencia de presentación de los newsgroup, definiciones de macro, keyboard remapping y otros comandos internos.

ARCHIVOS INTERNOS

⇒ *~/.nn/init*

Se encuentra la configuración personal. Cualquier personalización que se haya realizado queda especificado en este archivo y sobrescribe el archivo de inicialización sitewide.

⇒ *\$lib/init*

Es el archivo de inicialización sitewide utilizado por el administrador de news. Este es el primer archivo leído por nn.

⇒ *~/.nn/kill*

Este es el archivo kill.

⇒ *~/.newsrsc*

Es el archivo de inicio de news estándar. Este mantiene la información de los grupos subscriptos y de los artículos leídos en cada grupo.

NN al comenzar primero lee el archivo de inicialización systemwide, luego el archivo de inicialización personal.

Entre las acciones que puede tomar nn al leer estos archivos de personalización son:

- Comandos que se ejecuten, como por ejemplo limpiar la pantalla o realizar un cambio de directorio.
- Variables de texto o numéricas que tomen un valor específico, o boolean que tomen valor true o false.
- Al salvar un artículo, nn sugiere un nombre por defecto que es el nombre del newsgroups y en lugar de punto separado por /., se pueden especificar nuevos default.
- Setear comandos que pueden condicionalmente ser ejecutados usando comandos de grupo. Las condiciones pueden incluir shell commands, host names, terminal types y varias otras cosas.
- Se puede determinar el orden en que se muestran los newsgroup.

FINALIZACIÓN DE NN

Para salir se debe emitir el comando Quit tipeando la letra mayúscula "Q". Este comando actualiza los archivos de configuración, los registros de artículos leídos y realiza varias otras tareas de mantenimiento que acelerará las operaciones de nn la próxima vez que se lea un news.

También se puede salir de nn y mantener la configuración anterior a la sesión, para esto se debe tipear ":q!"

Estos dos comandos trabajan en los dos modos article-selected y reading.

TIN NEWSREADER

HISTORIA

Tin fue desarrollado por Lain Lea en Alemania, estuvo basado en el newsreader tass de Rich Skrenta y la primera versión fue publicada en 1991.

Ha sido regularmente mantenido y actualizado y es soportado por UNIX, AmigaDOS, OS/2 y Windows/NT.

En su momento, fue uno de los newsreaders más populares por las características prácticas que contiene y además por la interface elegante y amigable.

CARACTERÍSTICAS

- ⇒ Interfaz intuitiva y fácil de usar, esta simplicidad ayuda a aliviar la carga de soporte desde el news administrador.
- ⇒ Posee un sistema de archivo kill simple. Esta característica permite automáticamente eliminar de pantalla mensajes de poco interés y seleccionar artículos preferidos.

- ⇒ Posibilidad de agrupar artículos por subject.
- ⇒ La habilidad de etiquetas y operaciones sobre grupos de artículos. Grupos de artículos pueden ser guardados, ser impresos, ser enviados por mail, etc..
- ⇒ Ayuda en pantalla.
- ⇒ Incorpora facilidades de unpacking, decoding y unsharing postings.

COMIENZO

Existen varias formas de entrar a Tin, frecuentemente las más usadas son las siguientes:

Comienza tin usando el directorio local /usr/spool/news:

```
Tin
```

Comienza tin usando NNTP (network news transport protocol):

```
rtin o tin -r
```

Si es la primera vez que lee news o no leyó por mucho tiempo, se puede descartar los miles de mensajes acumulados en el sistema durante la ausencia con:

```
% tin -c
```

MODOS

⇒ *Group selection*

Este nivel navega entre los grupos subscriptos. Al seleccionar un grupo se entra al modo article-selection.

⇒ *Article selection*

Este modo navega entre los artículos no leídos y threads en el newsgroup. Cada thread es solo listado una vez. Desde este nivel se puede proceder al nivel thread-browsing o saltar directamente al modo article-browsing.

⇒ *Thread browsing*

Este nivel es similar al article-selection excepto que opera sobre thread. Este muestra cada artículo en un thread. En este nivel se puede seleccionar artículos individuales en un thread y proceder al nivel article-browsing.

⇒ *Article browsing*

Este es el nivel donde se leen los artículos que se seleccionaron. Las operaciones a nivel artículo, como salvar o responder mensajes, son realizadas aquí.

⇒ *Spooling directory selection*

Este nivel permite elegir diferentes directorios news-spooling.

SELECCIONANDO ARTÍCULOS

Al entrar a tin nos encontramos en el nivel group-selection, el cual lista varios newsgroups a los que se puede suscribir. Una pantalla típica de selección es la siguiente:

```
Group Slection (news.utdallas.edu 32)          h=help
-> 1      148 bit.listserv.christia
    2      3 dfw.eats
    3      dfw.eats
```

```

4      15 dfw.forsale
5      178 comp.lang.c++
6      49 comp.object
7      225 comp.lang.c
8      97 news.newusers.question
9      6 dfw.general
10     216 misc.jobs.offered
11      3 rec.humor.funny
12     256 soc.culture.asian.american
13      2 tx.general
14     13 tx.jobs
15     utd.acc.announce
16     utd.acc.downtime

```

```

<n>=set current to n, TAB=next unread, /=search pattern, c)atchup,
g)oto, j=line down, k=line up, h)elp, m)ove, q)uit, r=tpgggle all/unread
s)ubscribe, S)ub pattern, u)nsubscribe, U)nsub pattern, y)ank in/out

```

En la cabecera dice que nos encontramos en el nivel article-selection, el nombre del servidor de news y el recordatorio de ayuda. A continuación se despliega la lista de newsgroups subscriptos y un contador del número de artículos no leídos en cada uno.

En el pie de cada pantalla, tin siempre da un corto resumen de los comandos más importantes que se pueden usar en el nivel actual.

Para seleccionar un grupo se debe tipear su número y presionar la tecla "enter", nos podemos mover entre ellos con la tecla de flechas o usando la tecla "h" y "j" para movernos hacia arriba o abajo respectivamente. Al entrar a un grupo y ver los artículos para leer se presiona la tecla "enter" o la flecha derecha. También se puede utilizar Ctrol-F y Ctrol-B para pasar de pantalla hacia adelante o atrás respectivamente.

Cuando el grupo es seleccionado, se presenta la lista de artículos no leídos en modo article-selection. Una ventana típica es la siguiente:

```

                                Comp.lang.c++ (128T 178^ OK 11H R)                                h=help
-> 1 + char x Vs char x                                James Kanze
  2 + Union member with user-def = op. forbidden!    W Peter Jacobsen
  3 + Everything Virtual !!!!!                        Robert Martin
  4 + 3 Witty definition of your programing lenguaje Sami Nieminen
  5 + Project deadlines [8/7 is important!]          Raj Rao
  6 + Pointer to an array of objects                Andreas Balogh
  7 + 3 Class design: Member vs. Pointer            Robert Martin
  8 + Q: a way to use strings in switch-statement? Paul J Lucas
  9 + Static Inheritance                            Paul J Lucas
 10 * Forward declaration of typedef via "class T;" John Polstra
 11 * Default Argument if Arg is class object??    Edgar Holcomb
 12 + main ... :-)                                    James Kanze
 13 + argument conversion problem?                  Gnanasekaran Swam
 14 * 2 string class: is it part of ANSI C++?      Eric Roode
 15 + Multiply Overloaded Operators?                Ellster
 16 + How can I get a signal when data arrives on a Jon Olav Linde
121

```

```

<n>=set current to n, TAB=next unread, /=search pattern, ^K)kill/select,
a)uthor search, c)atchup, j=line down, k=line up, k=mark read, l)ist thread,
I)pipe, m)ail, o=print, q)uit, r=tpgggle all/unread, s)ave, t)ag, w)post

```

En la cabecera se muestra el nombre del grupo, el número de threads(128), los artículos no leídos (178), artículos killed automáticamente (0), artículos seleccionados "hot" automáticamente (11) y una indicación sobre si estamos viendo solo artículos no leídos (R). También se muestra un recordatorio del help.

A continuación se muestra la lista de artículos thread no leídos en el newsgroup. Cada línea muestra el número de thread, el estado ("+" significa no leído, "*" significa que el artículo es "hot", por ejemplo, automáticamente seleccionado), el número de artículos en el thread (blanco cuando solo es un artículo) y el subject y autor del primer artículo no leído en el thread.

Las teclas de navegación son las mismas que las utilizadas para la pantalla newsgroup-selection. Para seleccionar cualquier artículo o thread, se debe tipear este número y presionar la tecla "enter", o usar las teclas "up" y "down" o tipear "h" para moverse hacia arriba o "j" para moverse hacia abajo. Una vez seleccionado el artículo, presionar la tecla "enter" o la flecha derecha para mostrarlo. Se puede manejar tin solo con las tecla de flechas, con flechas "up" and "down" mueven hacia arriba y abajo en la corriente pantalla, mientras que "left" y "right" mueven a un nivel de menu hacia arriba y hacia abajo respectivamente. Presionando la tecla "Tab" se muestra el próximo artículo no leído.

En el pie de la pantalla se muestra un resumen de los comandos más frecuentemente usados en este nivel.

Finalmente se puede listar los artículos de un thread en particular seleccionando el thread y presionando lowercase "l" (letra ele).

BUSQUEDA A TRAVÉS DE SUBJECT

Permite buscar tanto hacia adelante como hacia atrás a través de líneas de subjects. La búsqueda es case-insensitive.

Cuando se está viendo un grupo en particular y nos encontramos en el nivel subject-line, es decir, no viendo un mensaje específico, se puede buscar por la línea de subject hacia adelante tipeando "/" y a continuación el texto a buscar.

Al realizar más de una búsqueda va dejando la historia de los textos buscados en el prompt.

Si la búsqueda se desea hacia atrás se debe utilizar "?" en vez de "/".

LEYENDO ARTÍCULOS

Al seleccionar un artículo para leer desde el modo article-selection o thread-browsing, nos encontramos parados en el nivel article-browsing, donde se puede ver el artículo seleccionado y realizar otras operaciones, como salvarlo, responder, etc.. El ejemplo siguiente muestra una típica pantalla en modo reading:

```
Sun, 12 Jun 1999 03:00:32      news.announce.newusersThread 14 of 21
Lines 816  Answers to Frequently Asked Questions      No responses
Netannounce@deshaw.com      Mark Moraes

Archive-name: usenet-faq/part1
Original-author: jerry@eagle.UUCP (Jerry Schwarz)
Comment: enhanced & edited until 5/98 by spaf@cs.purdue.edu (Gene Spafford)
Last-change: 23 Mar 1999 by moraes@deshaw.com (Mark Moraes)

      Frequently Submitted/Asked Items

This document discusses some questions and topics that occur repeatedly on USENET.
They frequently are submitted by new users, and result in many followups, sometime
swamping groups for weeks. The purpose of this note is to head off these annoying
events by answering some questions and warning about the inevitable consequence of
asking others. If you don't like these answers, let the poster of thie article know.

<n>=set current to n, TAB=next unread, /=search pattern, ^K)ill/select,
a)uthor search, B)ody search, c)atchup, f)ollowup, k=mark read,
I)pipe, m)ail, o=print, q)uit, r)eply mail, s)ave, t)ag, w=post.
```

La línea superior muestra varias piezas de información, el día y la hora en que el artículo fue postado, el grupo de noticias (news.announce,newsusers), el número de thread, el número de líneas en el posting (816), el subject (Answers to Frequently,...) y el número de respuestas de este artículo (none). La tercer línea identifica el autor (Mark Moraes).

Presionando la barra espaciadora o la flecha hacia abajo pasa de pantalla hacia adelante en el artículo; si estamos al final del artículo procede a mostrar el próximo artículo. Tipeando "b" o la flecha hacia arriba mueve la página hacia atrás y tipeando "n", la flecha derecha o la tecla "Tab" se pasa al próximo artículo; "p" mueve al artículo anterior. La flecha izquierda vuelve al artículo o al menu thread-selection.

Algunas veces se deseará volver atrás y ver un artículo o tópico en particular. Fácilmente se puede rellamar a mensajes en un newsgroup tipeando "r".

NAVEGANDO

Si se desea saltar directamente a un grupo de noticias particular se usa el comando "g" (goto). Este puede ser ejecutado desde cualquier nivel group-selection o article-selection. Al tipear "g", tin va a la línea de comandos para que se ingrese el nombre del newsgroup. Si se ha suscrito al grupo, el cursor es puesto en el grupo, en cambio, si no se está suscrito al grupo, pregunta donde gustaría que estuviera en la lista, seleccionando "\$" pone el newsgroup al final de la lista de newsgroup; tipeando un número significa la posición donde tomará lugar el grupo en la lista, el default es la posición corriente en la lista del grupo.

OBTENIENDO AYUDA

Tin provee dos caminos para obtener ayuda on line, primero un pequeño mensaje de ayuda aparece en la parte inferior de la pantalla. Si no se ve este mensaje, se debe tipear "H".

Cada nivel tiene varias pantallas de ayuda asociadas con él. Tipeando "h" se despliega la pantalla de ayuda del corriente nivel. El ejemplo siguiente muestra la primera pantalla de ayuda para el nivel group-selection.

```

Group Selection Commands (pag 1 of 2)

^D^U      Down (^U = up) one page
^F^B      Down (^B = up) one page
<SPACE>b  Down (b = back up) one page
jk        Down (k = up) one line (vi style)
4$        Choose group 4 ($ = choose last group)
g         Choose group by name
n<TAB>    Choose next group with unread news and enter it
N         Choose next group with unread news.
/?        Choose group by group name string forward (?= backward) search
          (all searches are case-insensitive and wrap around to all groups)
#         Choose range of groups to be affected by following command
<CR>     Read chosen group
w         Write (post article to) chosen group
cC        Mark all articles in chosen group read (C = and choose next unread)
z         Mark all articles in chosen group unread
^R        Reset .newsrsc so all articles are unread
Y         Yank in/out unsubscribed groups from .newsrsc
Y         Yank in active file to see any new news

PgDn, End, <SPACE>, ^D - page down. PgUp, Home, b, ^U - page up. <CR>, q - quit

```

EMERGENCIAS

Presionando la tecla "Escape" cancela cualquier operación que esté en proceso y retorna a cualquier cosa que estuviera haciendo previamente. Esto es muy útil si uno llegara a dudar que comando se tipeó o si accidentalmente se tipea un comando que no desea.

Tipeando CTRL-L se refresca la pantalla.

DECODIFICANDO ARTÍCULOS

Tin puede decodificar artículos que han sido posteados en diferentes formatos, estos incluyen multipart shar files y uuencoded binary files. También permite extraer automáticamente y listar tabla de contenidos de archivos zip.

Para leer los artículos encriptados se debe tipear el comando "d".

POSTEAR ARTÍCULOS

Hay dos caminos para postear artículos en tin. Usando el comando w(write) para un nuevo artículo o f(follow up) para responder el post de alguien. Se puede escribir el comando desde cualquier nivel, salvo la última opción que tiene que estar en modo browsing.

Al tipear el comando se abre el editor para escribir el texto del mensaje.

⇒ *Postear un nuevo artículo*

Tipear el comando "w" y tin pregunta por el subject

```
Post subject []> test post
```

Entonces da lugar al editor para escribir el texto del artículo. El header muestra el subject y el newsgroup. Se puede editar el contenido del header pero no se puede cambiar las letras del comienzo de la línea. Luego debe quedar una línea en blanco entre el header y el cuerpo del mensaje.

```
Subject: test post
Newsgroup: misc.test
Summary:
Keywords:

Test post
...
Mark Harrinson
```

Una vez que se termina de escribir el texto, se salva en un archivo temporal y se sale del editor. Entonces aquí se debe elegir entre las siguientes opciones:

- Quit; nada es posteado y el artículo se pierde
- Edit(re-edit) el posting. Entonces se vuelve al editor.
- Post the article: esta es la acción por defecto.

Check Prepared Article

```
Your article will be posted to the following newsgroup:  
Misc.test      For testing of network software, Very boring.
```

```
q)uit, e)dit, p)ost: p  
a)abort e)dit m)ail, r)eedit, s)end v)iew w)rite  
Action: (post article)
```

⇒ *Respondiendo un artículo*

Prácticamente se siguen los mismos pasos para responder el posting de una persona que para crear un nuevo artículo:

Tipear "f" para postear un follow-up de un artículo que se está leyendo. El texto del artículo al que se está respondiendo es incluido en la respuesta. Para que no se incluya se debe tipear "F".

El subject que toma es el del mensaje al que se está respondiendo.

ENVIANDO MAIL

Tin provee servicios básicos para enviar correo electrónico, se puede responder al autor de un artículo en particular o enviar a otras personas el artículo interesante.

⇒ *Reply*

Lo más común es devolver al autor de un artículo sin enviarlo al grupo entero.

Para enviar una respuesta se debe tipear "r" o "R" mientras se está leyendo el artículo (r incluye el texto original y R no lo incluye). Entonces tin abre el editor donde se puede

escribir el mensaje. Al salir del editor se da la opción de salir reeditando el mensaje o enviando este al destinatario.

⇒ *Enviar un post vía Mail*

Se puede reenviar un artículo via mail a personas que no tienen la posibilidad de leerlo, también es un camino para salvar un mail o dar un recordatorio a uno mismo enviándose un mail.

Se invoca al mail con el comando "m", el cual da el prompt con las siguientes opciones:

```
Mail a)rticle, t)hread, h)ot, p)attern, T)agged articles, q)uit.
```

Se puede elegir una de estas opciones descritas anteriormente, luego se da la dirección y se elige salir, editar o enviar al destinatario.

KILLING Y SELECTING ARTICLE

Al leer news, frecuentemente se desea buscar el artículo de un subject o autor en particular.

Tin hace fácil el auto-kill de los artículo poco interesantes u ofensivos y auto-select de aquellos interesantes. Esto puede ser hecho en un grupo en particular o a través de todos los grupos sobre una base permanente o temporal.

Para killiar un artículo, se debe tipea CTRL-K desde cualquiera de los niveles article-selection o article-browsing. Presionar CTRL-A para auto-select de artículos. Tin muestra una pantalla que permite especificar si se desea kill or select.

FINALIZACIÓN

Para salir en cualquier nivel de Tin, se debe emitir el comando quit tipeando la tecla "Q" en mayúscula o minúscula, tipeando "Q" mayúscula se sale de Tin y tipeando "q" minúscula toma el próximo nivel superior en la jerarquía de los newsreading.

Al salir, Tin actualiza los archivos de configuración, grava los artículos leídos y realiza otras tareas de mantenimiento.

FREE AGENT

HISTORIA

Forté Free Agent fue creado en California en la década del 90 y está desarrollada sobre un ambiente Windows.

CARACTERÍSTICAS

- ⇒ Permite configurar los paneles con las características que prefiera el usuario y parametrizar las barras de herramientas para que contengan las funciones más utilizadas por el usuario.
- ⇒ Soportado para Windows 95 y Windows NT – 32 bits. Agent soporta nombres largos como bien dial-up networking (windows 95) y Servicios de Acceso Remoto (Windows NT).
- ⇒ Soporta MIME, permitiendo mayor flexibilidad en el envío de mensajes y attachment.
- ⇒ Utiliza threading multi nivel, indentando respuestas para mensajes bajo el mensaje inicial, así se puede continuar una discusión sin tener que navegar buscando las respuestas. Agent provee un número de shortcuts para navegar eficientemente a través de todos los mensajes.
- ⇒ Se puede filtrar los mensajes que han sido posteados a múltiples newsgroups, con el fin de evitar que se lea el mismo mensaje más de una vez.
- ⇒ Permite reducir el desorden de grupo de noticias soportando filtros (también llamados Kill and Watch lists). Se puede killiar o watch messages con un especificado subject, autor or longitud del mensaje. También tiene las opciones Ignore y Watch, con las cuales permite saltar o recuperar mensajes en threads especificados.

-
- ⇒ Se pueden crear carpetas con cualquier tópicos para organizar la información, simplemente con drag and drop se mueven los mensajes a las carpetas personales, lo mismo se puede realizar con los mails.
 - ⇒ Trabajar en forma offline y online.
 - ⇒ Se puede enviar y recibir mensajes de mail y crear un address book para las direcciones de mails más frecuentemente usadas.
 - ⇒ Agent soporta una variedad de seteos de caracteres para diferentes lenguajes, incluyendo algunos con caracteres especiales acentuados como francés o con un único seteo de caracteres como ruso. Estos seteos de caracteres permiten ver los mensajes correctamente para el lenguaje dado.
 - ⇒ Luego de componer un mensaje, Agent permite realizar un chequeo de errores, para ellos cuenta con un número de diccionarios de lenguajes específicos.
 - ⇒ Permite manejar el propio espacio de disco diseñando estrategias de purging. (ejemplo, mantener los mensajes solo por un período de tiempo disponible en el server).
 - ⇒ Todos los URL en mensajes (direcciones web and direcciones de mail) aparecen con un link interactivo que puede ser lanzado con un simple doble click en ellos.
 - ⇒ Permite leer y escribir archivos newsrc.

COMIENZO

Al ejecutar por primera vez la aplicación despliega una pantalla pidiendo la siguiente información:

- *News (NNTP) server*: el nombre del servidor de news o su dirección IP.
- *Mail(SMTP) server*: el nombre del servidor de mail o su dirección IP.
- *Mail Address*: la dirección mail.

- *Full Name (optional)*: el nombre completo del usuario.
- *Time Zone*: Si la aplicación es de 16-bit hay que ingresar la zona horaria donde se esta localizado.

Provee un botón para obtener esta información desde los programas Netscape, News Xpress, Trumpet News, WinVN.

Si en el programa seleccionado tenía suscripción a grupos de noticias, estará suscrito en Agent también.

MODOS

Dependiendo de la actividad que se necesita realizar, Agent permite trabajar en forma online u offline.

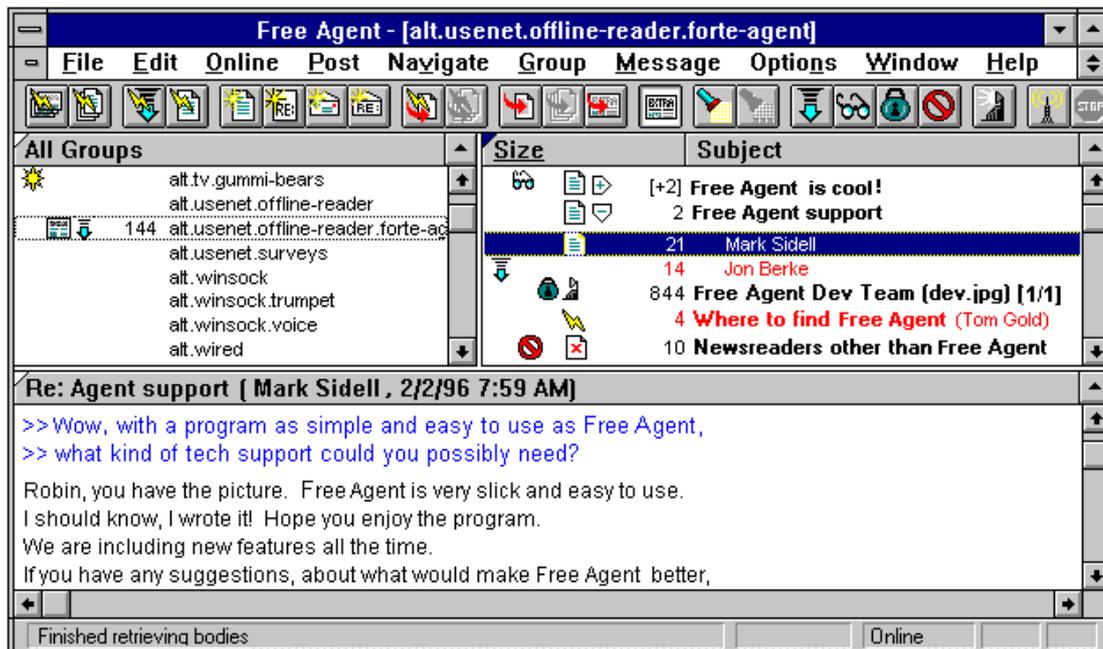
El modo online permite mantener conexión a Internet durante el tiempo que el programa está corriendo. Al correr Agent, este se conecta al server para ver cuantos mensajes nuevos hay en el grupo en el que está suscripto. Se puede entrar un grupo, Agent baja solo los headers de los artículos y se busca el que desea, pero los mensajes en si mismos no están en la máquina local, cuando queremos leer un artículo en particular, el contenido del mismo se obtiene del server, salvo que específicamente se haya guardado en local.

El trabajar offline permite una vez conectado al server, bajar todos los artículos nuevos para el grupo que está suscripto y desconectarse. Se pueden leer los mensajes cuando se desea, componer respuestas y hacer nuevos artículos para ser cargados la próxima vez que nos conectemos al servidor.

Se debe tener en mente que Agent mantiene en su disco local artículos y mensajes mail, no se debe trabajar online para revisar mensajes que ya han sido recuperados.

INTERFACE

La pantalla principal se encuentra dividida en tres paneles principales y barras especiales:



⇒ *Panel Lista de Grupos*

En este panel se muestra la lista de grupos de noticias y carpetas. Haciendo clic en el título del panel, se puede seleccionar grupos de noticias y carpetas que se desea ver: todos, suscriptos, nuevos o solo carpetas.

⇒ *Panel Lista de mensajes del grupo seleccionado*

En este panel se muestran los headers de los artículos del grupo o carpeta seleccionado y por cada uno de ellos el subject, fecha y hora, autor y el artículo es mostrado. Los thread se pueden ver expandidos con sus respuestas o no, si es expandido, todas sus respuestas son identadas siguiendo al artículo y solo el autor es mostrado ya que el subjects es el mismo que el mensaje original.

Se permite ordenar por los tópicos de las columnas, se mantiene subrayada la que corresponde al orden, por thread, por subject, por autor, fecha, tamaño.

⇒ *Panel Texto del Mensajes seleccionado*

Muestra el cuerpo del mensaje.

⇒ *Barra de Menú*

Se puede acceder a las opciones clicleando en el nombre del menu o presionando ALT más la letra del menú subrayada.

⇒ *Barra de Herramientas*

A través de los iconos de esta barra se obtiene un acceso rápido a los comando más utilizados en Agent. Se puede parametrizar con los iconos de los comandos que cada persona utilice con mayor frecuencia. Por cada botón aparece un hint descriptivo. Si dado un momento, la función de uno de los botones no está disponible, este aparecerá tenue.

⇒ *Barra de Estado*

Se utiliza para mantener información del estado actual a medida que se está navegando con Agent.

⇒ *Menu Contextual*

Muestra un menú con los comando más utilizados presionando el botón derecho del mouse.

Agent permite configurar los paneles con las características que prefiera el usuario y parametrizar la barra de herramientas que contenga las funciones más utilizadas por el usuario. También da la posibilidad de utilizar teclas rápidas y ver cualquiera de los tres paneles en pantalla completa. Otra facilidad visual es el manejo con colores de artículos leídos y no leídos. Agent utiliza iconos e indicadores al costado de un grupo o un header en los dos primeros paneles. Esto lo hace para tener una visión gráfica del estado de cada grupo o mensajes. Algunos de ellos indican: marcado para recuperar, grupo suscrito, nuevo grupo, cantidad de mensajes no leídos, thread expandido, thread colapsado, ignorar thread, offline, SCRL y otros. Se pueden ver los tres paneles al mismo tiempo o maximizar la información para ver la pantalla completa del panel que tiene el foco a través de la opción Zoom.

BÚSQUEDAS POR SUBJECTS, AUTORES Y MENSAJES

Agent permite que se realicen búsquedas de grupos por nombre y de mensajes por subject. Para buscar un grupo se debe tener el foco en el panel de grupos, luego elegir la opción "All Groups", elegir del menú Edit la opción "Find" y entrar una parte del nombre del grupo. Al presionar el diálogo de búsqueda " Find First" se encuentra el primer grupo en el panel de grupo que se corresponde con el patrón ingresado y posteriormente presionando la opción "Find Next" se van marcando los sucesivos grupos encontrados.

Para encontrar un mensaje por subject o autor se debe tener el foco en el panel de lista de mensajes, elegir del menú Edit la opción "Find", luego se debe ingresar en el dialogo de búsqueda parte del subject o del autor para encontrar los correspondientes mensajes y posteriormente presionando la opción "Find Next" se van marcando los sucesivos subject o autores encontrados.

SELECCIONANDO ARTÍCULOS

Buscar a través de lista de grupos es la tarea que el usuario realiza regularmente. Para encontrar un grupo en particular Agent tiene un entrada de menu Edit y a través de la opción Find permite realizar esta tarea.

Al encontrar el grupo de interés deseado y hacer un doble clic, este aparece y además se despliega un dialog box View Empty Group. Si se desea obtener más información sobre los temas tratados en el grupo antes de suscribirse a él se puede presionar el botón Sample Message Headers, sino con solo presionar el botón Subscribe se realiza la suscripción al grupo. Si se desea con posterioridad suscribirse o desuscribirse a un grupo, se puede realizar accediendo a la entrada de menu Group.

Desde la opción de menu Show flyout-Subscribe Group se permite elegir los Grupos Suscritos, entonces antes de desplegarse los mismos en el panel, este realizará un filtro de los grupos que son de interés del usuario, si en algún momento, se desea romper el filtro para ver nuevamente todos los grupos, se accede nuevamente a la entrada menú Show flyout –All Groups

Si se desea bajar todos los headers de todos los grupos al que el usuario está suscrito, se elige desde la entra de menú Online la opción Subscribed Group - Choose Get New Headers

Una vez que los headers son recuperados, se puede navegar por ellos para ver cual son de interés del usuario, si se está usando el modo online, con un simple doble clic o presionando la tecla Enter se obtiene el mensaje, en cambio, si se está utilizando el modo offline, se deberá marcar el mensaje, esto se realiza seleccionándolo y presionando "M" o eligiendo la opción Mark Retrieval desde la entrada de menú Message.

Para recuperar todo el texto de los mensajes marcados, se selecciona la opción Get Marked Message Bodies desde el menú Online.

LEYENDO ARTÍCULOS

Todos los headers de mensajes recuperados se verán coloreados en rojo, esto significa que el mensaje es nuevo o no fue leído.

A continuación se procederá a seleccionar el mensaje deseado como se explica en el paso anterior.

Cualquier mensaje que no haya sido marcado previamente será marcado automáticamente como leído.

Para leer un mensaje seleccione este y elija la opción View Selected Message desde el menú Navigate. También se puede leer un mensaje simplemente haciendo doble clic o resaltando este y presionando la tecla Enter.

Para leer el próximo mensaje no leído, usar los comandos View Next Unread Message (Ctrl+N) o View Next Unread Message in Thread (Ctrl+T) en el menú Navigate. Estos comandos saltan al próximo mensaje no leído y devuelve el cuerpo del mismo.

NAVEGANDO

Agent provee un rico seteo de herramientas para moverse entre newsgroup y mensajes. La mayoría de estas herramientas son accedidas desde la opción de menú Navigate, la barra de herramientas y el teclado. Es muy probable que una vez que se familiarice con el lector, el usuario no utilice todas las opciones disponibles, entonces se pueden elegir técnicas de navegación que mejor se adapten a sus necesidades.

⇒ *Moviéndose entre paneles*

Como ya mencionamos con anterioridad, Agent está compuesta por tres paneles: un panel de grupos, un panel de lista de mensajes y un panel de mensaje.

Se pueden ver los tres paneles al mismo tiempo o maximizar la información para ver en pantalla cada uno de los paneles.

Para maximizar la ventana se puede seleccionar Zoom Window desde el menu Navigate, presionar "Z" o hacer clic en el icono Maximize en el panel.

Para ver los tres paneles al mismo tiempo se selecciona Zoom Window desde el menu Navigate, presionar "Z" o hacer clic en el icono Restore.

Se puede ir moviendo el foco entre los paneles tanto con el mouse como con el teclado.

Para moverse con el mouse, cuando se están viendo los tres paneles, simplemente hacer clic en el panel deseado y si el panel se encuentra maximizado, hacer clic con la tecla tab para seleccionar el panel que se desea ver.

Para moverse entre paneles con el teclado, presionar la tecla tab o pulsar Shift+Tab o Escape para moverse en orden inverso. Otra opción es tipear las teclas "1", "2", o "3" para moverse de panel a panel. Presionando "1" da lugar al foco en el panel de Grupos, "2" da lugar al foco en el panel de Lista de Mensajes y "3" da lugar al foco en el panel de Mensajes.

⇒ *Moviéndose entre mensajes*

Esta es la acción más simple y más común que se realiza.

Para moverse al anterior o siguiente mensaje en el panel de lista de Mensajes se utiliza las teclas de flechas hacia arriba y hacia abajo, en el panel de Mensajes para moverse al anterior se elige la opción de menú Up to Previous Message en la entrada de menú Navigate y para ir al siguiente se elige la opción Down to Next Message o se presiona la tecla "D".

Para moverse entre mensajes no leídos se presiona la tecla "N" o el icono Skip to Next Unread Message en la barra de herramientas, si se quiere que además se muestre el cuerpo del mensaje presionar "B".

⇒ *Navegando y recuperando mensaje en Usenet al mismo tiempo*

Para recuperar el próximo mensaje no leído se presiona la tecla Ctrl+N o el icono de la barra de herramientas View Next Unread Message

Presionando Ctrl+T o el icono de la barra de herramientas View Next Unread Message in Thread salta al próximo mensaje no leído en el corriente thread y lo recupera.

⇒ *Usando una simple tecla para ver el próximo mensaje no leído*

En el tab de Navegación en el diálogo de General Preferences se puede marcar la opción Single Key Read.

Esto permite que se vea el cuerpo del próximo mensaje no leído con simplemente presionar la barra espaciadora, mientras que el panel de grupos y el panel de lista de mensajes tienen el foco.

⇒ *Expandir y plegar threads*

Agent despliega respuestas de mensajes identadas bajo el mensaje original. Al principio se muestra plegado y solamente el mensaje del tope es mostrado.

Para expandir un thread se debe hacer clic en el icono de thread o seleccionar el mensaje y presionar la tecla "+".

Para expandir todos los thread en el grupo, presionar la tecla shift junto con la tecla "+".

Para plegar un thread se debe hacer clic en el icono del thread o seleccionar el mensaje y presionar la tecla "-".

Para plegar todos los thread del grupo, se debe presionar la tecla "shift" junto con la tecla "-".

OBTENER AYUDA

A través de la entrada de menú Help o presionando la tecla F1, Agent realiza el despliegue de ayuda interactiva.

DECODIFICANDO ARTÍCULOS

Algunos mensajes de Internet contienen texto codificado con un esquema llamado ROT13. ROT13 codifica el texto para que no se puede leer sin una decodificación previa.

Para decodificar un mensaje ROT13 seleccionar el texto codificado y elegir la opción de menú Apply ROT13 o presionar la tecla "Ctrl+R."

Cuando se decodifica un mensaje recuperado, la decodificación solo afecta a lo que se muestra. Si más tarde el mensaje es vuelto a editar, nuevamente se verá el texto codificado originalmente.

POSTEANDO ARTÍCULOS

Agent permite realizar la publicación de mensajes en un grupo de noticias y responder a artículos posteados.

Para esto se deberá ingresar el nombre del grupo donde se desea publicar o los nombres de todos los grupos separados por coma en caso que sean varios los grupos destinos.

Se debe ingresar una pequeña frase para identificar el mensaje, lo que se denomina subject.

Como una entrada opcional se puede detallar una dirección de mail si se desea enviar una copia a personas directamente.

Si se está posteando una respuesta a un mensaje y se desea enviar un copia por mail al autor del mensaje original, se selecciona la dirección de mail de la lista.

Para enviar un mail a más de una persona se deben ingresar las mismas separadas por coma.

Otro punto opcional es la inclusión de attachment, si se desea incorporar uno al mensaje, se debe especificar el nombre del archivo y presionar el botón de Attachment, aquí se desplegará un browse para buscar el archivo.

Si se desea agregar una firma al final del mensaje se debe elegir la misma o seleccionar "none" en caso que no se desee agregarla. Se permite crear firmas por defecto, modificar las existentes y crear nuevas a través del cuadro de diálogo Signatures. Agent no muestra la firma en la ventana, esta es agregada al mensaje cuando se está transmitiendo.

Si se desea transmitir inmediatamente el mensaje a Internet se debe presionar el botón "Send Now", en cambio si se está utilizando Agent offline y se tiene mensajes que se desean enviar presionar "Send Later".

Una vez que se ha concluido la composición del mensaje se selecciona desde la entrada de menú Online la opción Post Usenet and Email Messages para transmitir el mensaje a Internet.

A través del botón "Save" se puede salvar un mensaje que se está componiendo y este aparecerá en el Outbox, para ver el Outbox seleccionar desde el menú Window la opción Open Outbox, aquí se mostrarán todos los mensajes pendientes, haciendo doble clic sobre ellos se pueden editar para su envío.

ENVIANDO MAIL

Con Agent se puede enviar mails y responder artículos a través de ellos.

Para responder o reenviar un artículo con mail se debe seleccionar el mensaje al cual se desea responder o reenviar, luego se debe abrir una nueva ventana de Posting y desde el menu "Post" elegir una de las opciones "New Email Message" , "Reply via Email" o "Forward via Email" dependiendo del tipo de mail que se desea enviar.

A continuación se deben rellenar todos los campos que solicita la ventana de Post de acuerdo a la opción elegida, también se pueden agregar archivos attachados.

Por último, si se está trabajando en modo online, al presionar la opción "Send Now" se transmite inmediatamente el mensaje a Internet, en cambio, si el modo de trabajo es offline y no se desea enviar inmediatamente el mensaje, se selecciona la opción "Send Later".

Una vez que estén todos los mensajes confeccionados, se elige la opción "Post Usenet" y "Email Messages" desde el menú Online para que los mismos se transmitan a Internet.

KILLIAR Y PURGAR

Agent permite automáticamente ignorar mensajes basados en subject, autor o longitud del mismo.

En general, automáticamente deletea o "purges," los mensajes viejos de Usenet.

Hay tres momentos en los que se puede purgar mensajes:

- Cuando se están recuperando nuevos headers: Agent puede purgar mensajes viejos de un newsgroup siempre que este recupere nuevos header de mensajes para el newsgroup.
- Cuando se cierra el lector: se puede purgar mensajes viejos de todos los newsgroups.
- Bajo demanda: Se puede manualmente purgar mensajes viejos seleccionando del menú Group la opción "Purge Newsgroups flyout", o borrando inmediatamente un mensaje, sin purging de otros mensajes en el grupo seleccionando, del menú Message la opción Delete Message.

Agent provee muchas opciones para decidir cual de los mensajes de USENET se purgan automáticamente, se puede purgar mensajes una vez leídos o luego de un período de tiempo.

A pesar del seteo de purgar automáticamente, se puede prevenir que algún mensaje en particular no se elimine aunque pertenezca al grupo seteado, para esto se debe resaltar el mensaje y elegir del menú Message la opción "Keep Message".

PERSONALIZAR

Una poderosa característica de Agent es la posibilidad que brinda de personalizar propiedades. Se pueden setear propiedades para que afecten a todos los grupos o solo a los grupos seleccionados.

ARCHIVOS INTERNOS

Agent permite leer y escribir archivos newsrc, así se puede mover información entre máquinas (por ejemplo es útil para una persona que utiliza el mismo lector en la casa y en el trabajo)

También se puede utilizar al cambiar de Lector de Noticias para no volver a realizar nuevamente la suscripción a los newsgroups favoritos.

Para exportar se selecciona la opción de menu File- Newsrc - Export, luego aparece un diálogo box donde se especifica el nombre y destino del archivo newsrc que será creado. Si se desea solamente exportar los grupos Suscritos, se debe marcar una casilla box.

Para exportar seleccionar File-newsrc-Import y se despliega un dialogo box para seleccionar el archivo newsrc a importar (localización y nombre).

Se puede configurar Agent para que automáticamente importe un newsrc al iniciar una aplicación y exporte al finalizar el mismo.

MULTITAREA

Agent permite hacer varias tareas simultáneamente, por ejemplo, mientras se están bajando los headers de un grupo , se puede leer mensajes de otro grupo. Si Agent se encuentra trabajando con una tarea online, la próxima acción a iniciar será encolada y Agent procederá a realizar esta cuando finalice la anterior, con la barra de estado se va mostrando las tareas encoladas e indica que tarea se encuentra corrientemente trabajando.

Se puede indicar que a ciertas tareas se le de prioridad más alta, en particular se le puede decir a Agent que recupere el cuerpo de un artículo aunque esté ocupado con otras operaciones online.

Es posible comenzar a leer mensajes nuevos de un grupo mientras se está recuperando los headers de otro.

FINALIZACIÓN

Para salir, se accede al menú File y a la opción exit, luego Agent actualiza automáticamente sus archivos de configuración.

CARACTERÍSTICAS A CUMPLIR POR EL LECTOR DE NOTICIAS

Una vez concluida las investigaciones preliminares sobre algunos programas lectores de noticias existentes en el mercado, hemos definido un conjunto de pautas básicas y necesarias a cumplir en el diseño de un lector, para proveer tanto la funcionalidad básica del mismo, como así también, ofrecer facilidades de uso.

A continuación se mencionan las características que cumple nuestro lector de Noticias:

- Mostrar todos los grupos de noticias dando la posibilidad de acotar la búsqueda: es provisto dotando a la aplicación de tres pestañas mediante las cuales el usuario puede elegir trabajar con: todos los grupos de noticias disponibles en el servidor, solamente los suscritos o los grupos que se hayan creado después de la última conexión.
- Mostrar todos los artículos presentes en un grupo permitiendo limitar la cantidad mostrada: es posible determinar la cantidad de headers de artículos que se recuperan. Esto se indica dentro de la configuración del usuario, aunque es posible en tiempo de ejecución recuperar todos los headers de un grupo.
- Proveer alguna forma de eliminar la obligatoriedad que el usuario tenga que leer todos los artículos del grupo para obtener el que es de su interés: los headers de los artículos son mostrados de distintos colores para distinguir entre los artículos que han sido recuperados, los que están aun sin leer y aquellos que son nuevos, es decir, se han publicado después de la última conexión.
- Facilitar el seguimiento de las respuestas de un artículo: las repuestas a los artículos son mostradas debajo del artículo original, en forma de árbol jerárquico, facilitando de ésta manera el seguimiento de los threads.
- Proveer distintas formas de búsqueda: es posible hacer una búsqueda sobre los datos que forman el header del artículo.
- Composición y edición de artículos: se implementó un editor para el manejo de la edición de artículos.

- Envío de respuestas vía mail: se utiliza el editor de los artículos para esta función.
- Proveer mensajes de error cuando surgen problemas de comunicación con el servidor de noticias.

Además de estos objetivos, otro punto que decidimos priorizar, fue la reducción del tiempo de conexión al servidor de noticias, dotando a nuestra aplicación de una base de datos local, a través de la cual, se permite mantener la información recuperada por el usuario entre sesión y sesión. Para este punto se ha aprovechado la posibilidad que nos ofrece Java de desarrollar aplicaciones que se ejecuten fuera de un browser Web, extendiendo no solo su portabilidad sino también su programabilidad.

AMBIENTE DE TRABAJO

JAVA

Una de las razones por las cuales Java ha sido promovido como un paso revolucionario en programación, es porque además de ser muy útil como lenguaje de programación tradicional, es importante ya que provee soluciones para la programación en la World Wide Web y ha surgido como una tecnología necesaria para cumplir las nuevas demandas de la red. Dicho de un modo sencillo, Java da vida a los sitios Web, pone interacción y lógica en páginas que, de otro modo, serían estáticas.

Java colma uno de los sueños mas acariciados por el creador de software: el de poder escribir dicho software en un solo lenguaje susceptible de ser admitido para trabajar en ventanas y en redes en las principales plataformas de sistemas operativos, sin necesidad de adaptaciones especializadas ni de una recompilación posterior. Pero los atractivos de este lenguaje no se agotan en esto, sus creadores se propusieron desarrollar con Java un lenguaje y un sistema en tiempo de ejecución simples, compactos, orientado a objetos, ampliables y seguros.

Java fue diseñado con múltiples niveles de seguridad incorporados en el compilador, el sistema de tiempo de ejecución y en los navegadores que admiten Java. Tales medidas de seguridad hacen que Java sea de por sí, mucho mas seguro a la hora de crear cualquier tipo de software de aplicación fiable.

El soporte para trabajo en redes que incorpora Java, así como la capacidad de su sistema de tiempo de ejecución para cargar dinámicamente bytecodes Java por medio de la red, lo hacen especialmente adecuado para las aplicaciones distribuidas de conexión en red. Proporciona asimismo, la capacidad de utilizar dinámicamente nuevos contenidos y software de gestión de protocolos.

Su carácter orientado a objetos combinado con sus numerosas comprobaciones de la integridad del tiempo de compilación y ejecución, eliminan de raíz muchos errores de programación difíciles de detectar. Por otra parte, el lenguaje Java prescinde por completo de muchas posibilidades de programación peligrosas, tales como, los punteros modificables, la conversión de tipos no comprobada y la comprobación débil de límites, que suelen permitirse en otros lenguajes de programación.

La API de Java proporciona pleno soporte a la programación multithread. Los programas multithread pueden ser desarrollados de una forma única y consistente sin depender de las particularidades caprichosas de la interfaz del sistema operativo principal.

Las clases y objetos Java admiten directamente los conceptos orientados a objetos de encapsulación, herencia, mensajes y métodos, y de ocultación de datos. Las interfaces Java admiten la herencia múltiple y el polimorfismo. El lenguaje Java tiene en suma, todas las ventajas de la programación orientada a objetos, pero sin las repercusiones de rendimiento que se asocian a los lenguajes puramente objetuales.

La API de Java presta también soporte al desarrollo de interfaz gráfica de usuario y al trabajo en ventanas, sin las complicaciones inherentes al mantenimiento de múltiples bibliotecas de clase de ventanas. El lenguaje cuenta además con varias herramientas de programación visuales creadas especialmente para Java.

La desventaja que tiene Java es la velocidad, como son programas interpretados nunca alcanzan la velocidad de un ejecutable.

AMBIENTES DE PROGRAMACION JAVA

Inicialmente comenzamos investigando JDK 1.1.3, el cual no proveía ambiente de desarrollo gráfico ni la posibilidad de desarrollar interfaces gráficas y de características similares a los programas utilizados por la mayoría de los usuarios hoy en día.

Continuamos investigando los lenguajes de programación Microsoft Visual J++ 1.1 e IBM Visual Age 1.0, los cuales si bien proveían ambiente de desarrollo gráfico, debugger y el resto de las características comunes de los lenguajes visuales, no incorporaban la posibilidad de desarrollo de interfaz iconizada con barra de menús, botones con gráficos, etc.. Para subsanar este inconveniente intentamos desarrollar un híbrido que contenía la funcionalidad desarrollada con uno de los dos productos mencionados y la interfaz gráfica la desarrollamos con la herramienta de creación visual NetObject BeanBuilder 1.0., la cual permite exportar el código generado de modo de poder utilizarlo como una nueva clase en los ambientes de programación. Si bien de esta forma se solucionaba la carencia gráfica de los lenguajes elegidos, se llegaba a un producto de baja performance.

Finalmente decidimos migrar nuestra aplicación a Borland Jbuilder 2.0, que provee un ambiente de desarrollo amigable, con incorporación de componentes SWING para la implementación de una interface gráfica adecuada.

BORLAND JBUILDER

Borland Jbuilder es un entorno integrado de desarrollo (IDE) en el que se puede editar, compilar, probar y depurar programas para un rápido desarrollo de aplicaciones, los cuales luego pueden funcionar sobre cualquier plataforma hardware.

Jbuilder crea soluciones Java absolutamente genuinas, y es perfectamente compatible con Java 1.1.

Incluye nuevas características como clases internas, archivos Java (JAR), conectividad de Java con Bases de Datos (JDBC), y tanto las Clases Fundadoras de Java (JFC) como el modelo de eventos del Kit Abstracto de Herramientas de Ventanas (AWT).

Jbuilder también incorpora JavaBeans, la arquitectura de componentes de Java. Estos Beans permiten aprovechar pequeños fragmentos de software reutilizables para crear aplicaciones cómodamente en lugar de construir programas desde cero. Con JavaBeans se podrá integrar fácilmente piezas de distintas fuentes para crear programas completamente funcionales.

Jbuilder permite crear programas de Java rápidamente, bien de forma manual o bien con la ayuda de los muchos asistentes, o incluso con el ratón.

Con la ayuda de los Asistentes (Wizards) y una gran galería de objetos, Jbuilder ayuda a crear applets para incluir en páginas Web o aplicaciones Java independientes.

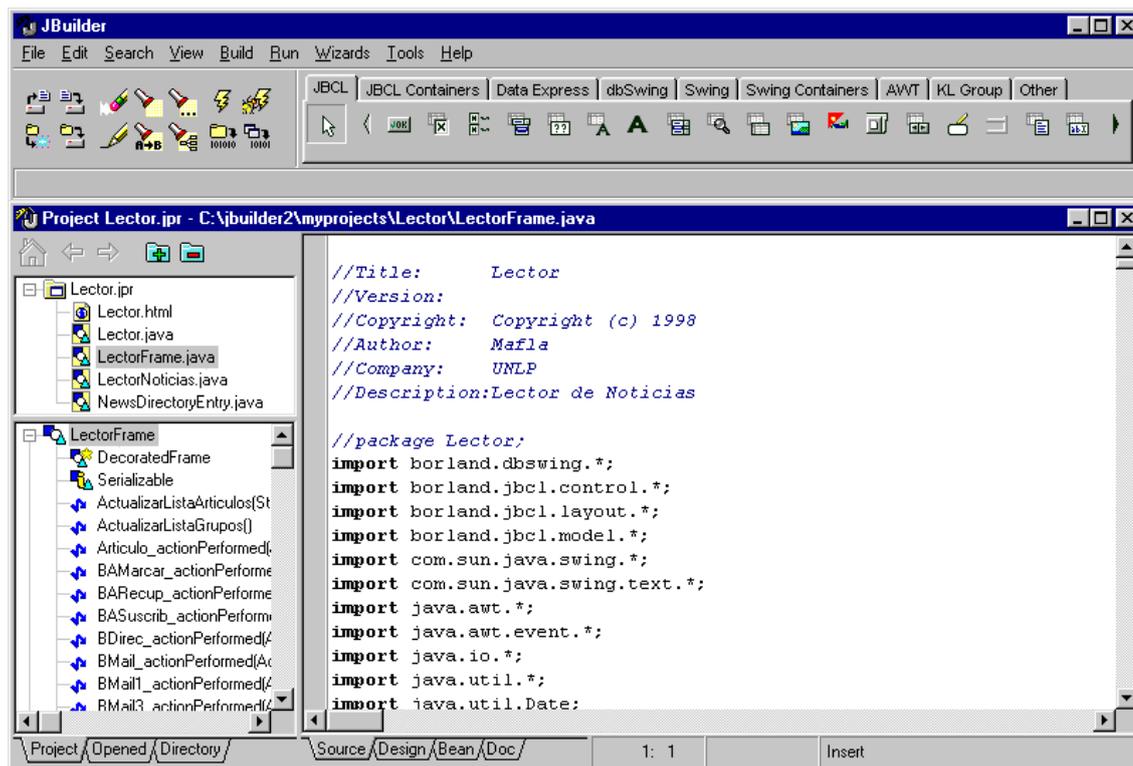
Los programas que se crean automáticamente con Java pueden ejecutarse sobre distintas plataformas. Cuando se compila un programa, el compilador crea archivos .class que contienen código (llamado bytecode) independiente de la plataforma en la que se ejecute, pues la Máquina Virtual de Java (Java Virtual Machine o JVM) lo comprenderá allá donde lo encuentre. Esta es una de las grandes ventajas de Java, una vez que escriba algo, podrá ejecutarlo donde quiera.

ENTORNO DE DESARROLLO

El entorno de desarrollo de JBuilder guarda cierta semejanza con el de las otras herramientas de Borland como Delphi y C++.

Cuando Jbuilder comienza, muestra el Entorno de Desarrollo Integrado (IDE) que es donde se construirá y probará los programas de Java.

El entorno de Jbuilder está dividido en dos zonas y cada una contiene a su vez varias partes:



La zona inferior contiene el espacio de trabajo principal que es el visor de aplicaciones. Este contiene tres zonas o paneles diferentes: el panel de navegación, el de contenido y el de estructura.

El panel de navegación, es donde se puede desplazar entre los distintos archivos de un proyecto, al cambiar de archivo, los otros paneles del visor de aplicaciones se actualizan para reflejar su localización. Como característica añadida, existe un icono Home que siempre nos devolverá al nivel superior del proyecto, en caso que se pierda.

El panel de contenido inicialmente muestra el código fuente del archivo que se encuentre seleccionado, sin embargo, es también el área de trabajo donde se depositan los componentes arrastrados, desde la paleta de componentes, con la ayuda de las lengüetas bajo el panel de contenido, se puede cambiar de una vista a otra.

El panel de estructura mostrará la estructura del archivo en forma de árbol cuando el elemento que esté seleccionado en el panel de navegación sea un archivo fuente .java.

Al minimizar una ventana del visor de tareas Jbuilder, se alinean estas ventanas en la parte inferior del área de trabajo de la pantalla mostrando solo su título, este procedimiento resulta útil cuando se trabaja con varias ventanas de visores de aplicaciones abiertas.

CREACIÓN DE UN PROGRAMA DE JAVA

Hay dos tipos de programas que se pueden crear con Java y Jbuilder. Los programas de Java que se ejecutan fuera del visor se llaman aplicaciones y las applet son programas de Java que se ejecutarán en páginas Web. Los applets no tienen un tamaño menor, sino solo se diferencian de las aplicaciones en que su seguridad es restringida. Normalmente una applet no puede acceder a la unidad local de disco duro, y solo se puede comunicar con la máquina donde se originó.

ASISTENTES

Hoy en día todos los entornos de desarrollo proporcionan asistentes u otras ayudas que hacen que raramente tengamos que comenzar una aplicación desde cero.

Los principales asistentes en Jbuilder son:

⇒ *Asistente de Proyectos*

A través de este asistente, el usuario puede crear un nuevo proyecto

En Jbuilder todo trabajo se incluye dentro de un proyecto. Si comienza la ejecución de cualquier asistente de Jbuilder sin un proyecto abierto, Jbuilder le recordará que debe crear uno antes con el Asistente de Proyectos, para continuar luego con el asistente que hubiera solicitado.

El asistente crea automáticamente dos archivos, uno `nombreproyecto.jpr` y otro `nombreproyecto.html`, este último contendrá una copia de la información que proporcionó al Asistente de proyecto.

⇒ *Asistente de Applets*

Permite al usuario crear una applet, programas que se ejecutan en un navegador, a través de tres pasos, primero, se debe dar un nombre al applet, luego establecer parámetros de ejecución y por último decidir si desea que se cree una página HTML.

El asistente crea automáticamente tres archivos nombrepquete.html, nombreclase.html y nombreclase.java.

⇒ *Asistente de Aplicaciones*

Este asistente permite crear aplicaciones que se ejecuten por si solas. De esta forma se puede automatizar la creación de una barra de menús, una barra de herramientas, una barra de estado o una caja "about", lo que no se podría hacer desde un applet ejecutándose en forma independiente.

El asistente crea automáticamente dos archivos nombreframe.java y nombreclase.java.

⇒ *Asistente de Anulación de Métodos*

Anular un método significa que hay un método de los que una clase ha heredado que deseamos modificar. Por ello, es necesario volver a escribirlo. Este asistente nos ayudará a realizar esta operación.

⇒ *Asistente de Implementación de Interfaces*

Permite escoger el nombre de la interfaz de una lista y añade el código necesario, de forma que nosotros sólo tengamos que rellenar la lógica que deseamos que realice. El asistente es realmente útil cuando una interfaz define varios métodos, pues no tendrá que escribir cada uno a mano.

⇒ *Asistente del modelo de datos*

Este asistente simplifica el acceso a datos desde la aplicación.

⇒ *Asistente de interacciones*

Ayuda a conectar beans generando automáticamente todo el código necesario para realizar la conexión.

LOS COMPONENTES Y LA LIBRERÍA

Los componentes que incluye un entorno de desarrollo definen muy a menudo hasta dónde se puede llegar con una herramienta, dado que las tareas que podemos llevar a cabo utilizando un componente suelen ser difíciles de realizar comenzando desde cero. JBuilder ofrece bastantes componentes predefinidos.

Además de los componentes de la librería estándar, como etiquetas, botones de radio, etc., también se incluyen algunos diálogos estándar, para abrir y guardar archivos, escoger un tipo de letra, etc.. Borland ha licenciado algunos componentes del KL Group, incluyendo algunos que mejoran a los controles estándar de la AWT.

Borland ha incluido también una serie de componentes propios, que componen la JBCL: entre ellos, los más interesantes son los que permiten llevar a cabo el acceso a bases de datos, ubicados en la página Data Access de la paleta. La información manejada por estos componentes puede ser editada o mostrada al usuario final a través de una serie de controles capaces de trabajar en colaboración con los componentes de acceso a bases de datos, ubicados en la paleta Controls, que también se pueden utilizar independientemente.

Aparte de los componentes visuales, JBuilder también proporciona, por supuesto, todas las clases del JDK de Java, y la librería JGL (Java Generic Library), de ObjectSpace, que contiene un gran número de clases y algoritmos de utilidad, como diccionarios, etc., basada en la filosofía de la STL de C++.

ACCESO A ARCHIVOS

Java proporciona acceso al sistema local de archivos a través de las clases del paquete java.io, que le permitirán leer y escribir bytes, caracteres u objetos, y compartir los resultados con cualquier otro sistema de Java en tiempo de ejecución con independencia de los formatos de datos específicos de cada plataforma.

El acceso al sistema local de archivos en Java está restringido, ya que los applets normalmente no tienen capacidad para ello sino que se hace a través de una aplicación.

LOS JAVABEANS

El uso de componentes reutilizables surge con la maduración de la industria del software.

JavaBean es un componente reutilizable de software que puede ser manipulado visualmente mediante una herramienta constructora.

Builder no solo es capaz de utilizar JavaBeans, sino también de crearlos.

Un bean se caracteriza por su estructura, de modo que una herramienta constructora puede descubrir todo lo que necesite acerca de él, siguiendo ciertas reglas de diseño.

Los bean han de saber como gestionar los eventos, es decir, han de ser capaces de reaccionar cuando ocurre algo. Un bean también ha de ser configurable visualmente; si no, la herramienta constructora no funcionará igual de bien.

Los bean también han de poder guardarse y recuperarse asimismo, para, de este modo, seguir funcionando a lo largo del tiempo, tras retomarse la ejecución de una aplicación previamente interrumpida.

⇒ *Propiedades, métodos y eventos*

Dentro de un bean, lo primero que hay que examinar son sus propiedades, es decir, sus atributos. Para leer o modificar el valor de una propiedad, se emplean un par de métodos con un nombre especial. Dentro de la herramienta constructora, un editor de propiedades permite al programador de la aplicación modificar las propiedades.

A continuación nos encontraremos con los métodos. Los métodos proporcionan el medio de manipular un bean y de acceder a sus propiedades.

Una herramienta constructora tendrá acceso a todos los métodos públicos de un bean, o sólo a unos cuantos si decidimos restringir el acceso.

Por último vemos los eventos, mecanismos de comunicación entre beans. Si un bean quiere saber cuándo ha ocurrido un evento, se registra a sí mismo en su fuente u origen, y así cuando el evento ocurra, la fuente del mismo notificará a cualquiera que esté registrado o escuchado.

⇒ *Personalización e introspección*

Cuando creamos un bean hacemos públicos sus detalles, es decir, los nombres de sus propiedades y eventos. Este proceso se denomina personalización (customization). Cuando un bean es utilizado por una herramienta constructora, ésta ha de conocer la información que hemos publicado, y el proceso mediante el cual la extrae del bean se denomina introspección.

⇒ *Persistencia*

Para que sean útiles, los bean han de disponer de algún mecanismo de almacenamiento. Si no tuviera la propiedad de persistencia, un bean presente en veinte millones de aplicaciones tendría el mismo aspecto en todas ellas, pero gracias a la persistencia podemos

personalizarlo para que sea distinto. Después de configurarlo, será guardado separadamente para cada aplicación.

⇒ *Jbuilder y los javabeans*

Jbuilder es una herramienta constructora. Ayuda tanto a crear beans como a manipularlos visualmente.

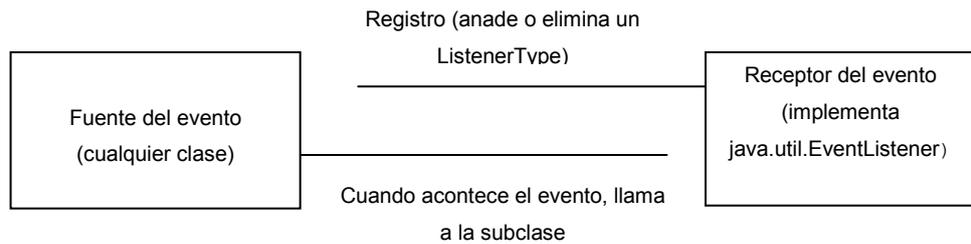
Jbuilder incluye un asistente llamado Asistente de Interacciones (Interaction Wizard), que ayuda a conectar beans generando automáticamente todo el código necesario para realizar la conexión. Además de añadir este código, el Asistente de Interacciones tiene incorporadas ciertas capacidades adicionales: el Editor del Asistente de Interacciones y el Ayudante de Expresiones. A través del editor del asistente de interacciones se restringe la lista de eventos y respuestas de un bean limitando la visualización a sólo los mecanismos de respuesta a eventos que más se use y como ayuda al editor del Asistente de Interacciones, el Ayudante de Expresiones le permite configurar ciertas opciones que influirán en la conexión.

⇒ *Configuración de la paleta de componentes*

Se puede ampliar la paleta de componentes de modo que pueda disponer de aun más beans. Ampliar la paleta es útil en el caso que se quiera usar con más facilidad beans que se hayan adquirido o creado.

⇒ *Eventos*

Los eventos o sucesos sirven como mecanismos de comunicación entre distintos beans. Si un bean quiere saber cuando sucede un evento, se registra a sí mismo como origen de ese evento. Así, cuando suceda informará a cualquiera que le interese. El evento debe ser una subclase de `java.util.EventObject`, mientras que ese "oyente interesado" implementa la interfaz `java.util.EventListener`. Todos los eventos de AWT funcionan del mismo modo, por lo tanto implícitamente todos los componentes AWT son beans. Todo este proceso es el que intenta esquematizar la siguiente figura:



⇒ *BeansExpress*

BeansExpress forma parte de Jbuilder Professional. Se puede encontrar en una lengüeta de la galería de objetos y ofrece cinco asistentes. Dos de ellos, OK Cancel Bean y DB Bean, son ejemplos completos de beans. Los otros tres, New Bean, Bean Info y New Event Bean, ayudan a crear beans nuevos.

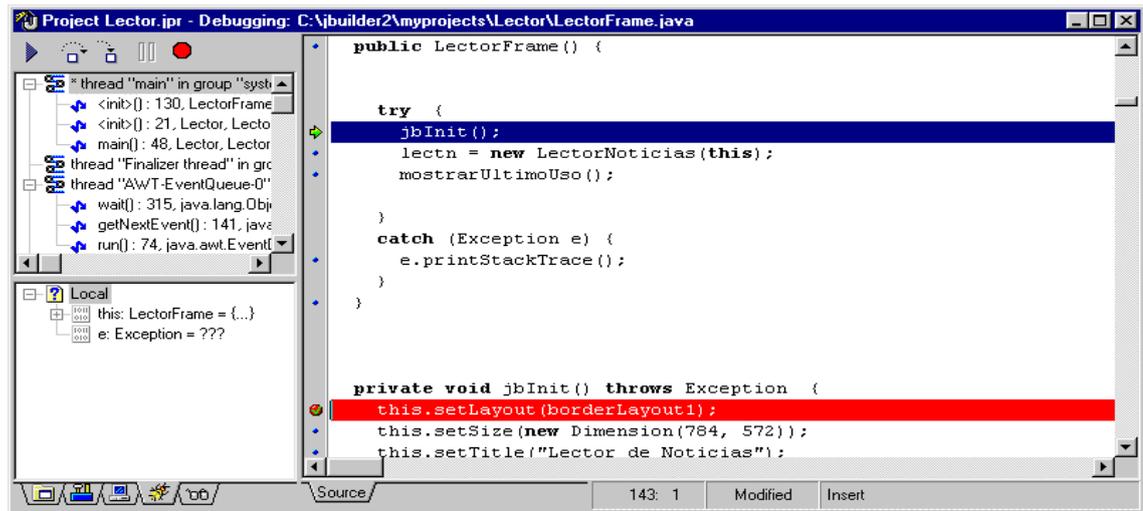
DEPURADOR DE JBUILDER

JBuilder ofrece un excelente soporte para la depuración: como suele ser habitual en todas las herramientas de programación hoy día, ofrece la posibilidad de utilizar puntos de ruptura (breakpoints), trazas a través del código, ver el contenido de las variables en tiempo de ejecución, etc.. Además, JBuilder soporta la depuración de programas con múltiples hilos de ejecución y dispone también de una ventana especial para ir registrando diversos sucesos, como la carga de una clase, la creación de un thread y su terminación, etc.: esta ventana es accesible mediante la opción de menú View|Execution Log.

Otra posibilidad interesante es la de desactivar la traza dentro de ciertos archivos, por ejemplo, los de la librería estándar, con lo que al realizar una traza nunca entraremos en su código fuente, lo que hace esta actividad mucho menos engorrosa: por supuesto, es posible especificar qué archivos no deseamos que sean trazables. Es posible acceder a la ventana que controla el funcionamiento de esta característica mediante la opción de menú View|Loaded classes.

La primera fase del proceso de depuración sería añadir sentencias `System.out.println` por todo nuestro código, para comprobar el estado del sistema en distintos puntos. Sin embargo, en vez de modificar el código, podemos usar el depurador para revisar la ejecución del programa.

Al comenzar el proceso de depuración cambiará el aspecto del visor de aplicaciones (AppBrowser).



Cuando se ejecuta un programa en modo de depuración se detiene en la primera línea ejecutable del código y el panel de threads y pilas (parte superior izquierda) muestra la lista de threads que están siendo ejecutados, además que es lo que están ejecutando, mediante una pila de llamadas.

El panel de datos (abajo a la izquierda) muestra las variables visibles en cada momento, y el panel de código (a la izquierda) muestra el código.

A la izquierda del panel de código se verá una barra vertical gris, en esta barra se señalan las líneas de código con puntos azules, la línea que se está ejecutando aparece con una flecha verde, y las líneas de comentario aparecen sin nada. Los puntos de ruptura se indican mediante puntos rojos marcados; son los puntos donde indicamos al depurador que la ejecución ha de interrumpirse.

El icono ejecución (Run) sirve para ejecutar el programa desde el punto en el que está hasta que encuentre algo que lo interrumpa. El icono seguir (Step Over) hace que se ejecute un método directamente, sin meternos dentro de él.

Paso a paso (Trace into) nos permite ejecutar el código línea por línea, entrando en los métodos cuyo código fuente esté disponible. Si no desea ejecutar paso a paso algunas bibliotecas, puede desactivar paquetes enteros mediante el comando Loaded Classes del menú View.

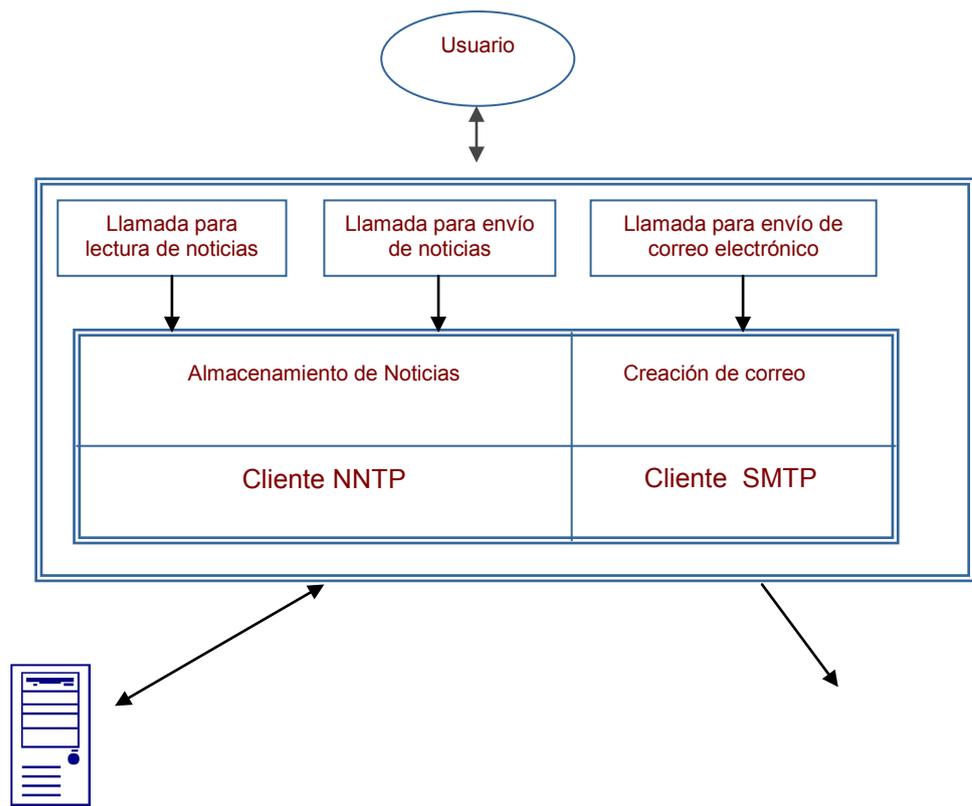
Para interrumpir la ejecución de un programa mientras está ejecutándose, se debe pulsar el icono Pausa. El icono Reiniciar interrumpe el programa y permite volver a ejecutarlo desde el principio.

APLICACION LECTOR DE NOTICIAS

ARQUITECTURA

Los sistemas de Noticias están compuestos por dos componentes principales: un lector de noticias y un servidor de noticias. Los lectores de noticias permiten a los usuarios redactar y enviar artículos y recuperarlos de los servidores de noticias. Estos últimos almacenan y permiten el intercambio de artículos por una red o grupo de redes.

El protocolo NNTP es utilizado por los dos componentes para enviar y recibir información USENET a través de una red TCP/IP. Las noticias de USENET utilizan dos variantes del protocolo NNTP: una para la comunicación entre servidores adyacentes y la otra para la comunicación entre el lector de noticias y un servidor.



La aplicación provee una interface estándar para el manejo de correo electrónico y noticias. Permite la composición de artículos y correo, y la lectura de noticias.



Realiza las siguientes funciones, que constituyen el manejo básico para un cliente de noticias:

- Crea un artículo consistente de una colección de atributos que constituyen el header del mismo y un bloque de datos.
- Envía un artículo a un grupo de noticias o a un receptor en particular.
- Crea una sesión de transporte para el envío de noticias.
- Crea una sesión de transporte para el envío de correos.
- Recupera un grupo de noticias y artículos desde el servidor de noticias.

COMPONENTES PRINCIPALES

⇒ *Clientes NNTP y SMTP*

Modeliza el agente de transporte desde el servidor de noticias, hasta el usuario y viceversa, corresponde a la implementación del protocolo NNTP para el intercambio de mensajes.

Utilización de la API JavaMail para el envío de mensajes de correo electrónico.

⇒ *Grupo de Noticias*

Modeliza el grupo de noticias, indicando la cantidad de artículos que contiene, el primer y último artículo del grupo, si está suscrito, si el grupo ya fue recuperado de la red y grabado en la base de datos local.

⇒ *Artículo*

Implementación del artículo propiamente dicho incluyendo la información y mantenimiento del header (subject, autor, fecha, cantidad de líneas, etc.) y el cuerpo de un artículo una vez que este ha sido recuperado, como así también se define el control de lectura de los mismos.

⇒ *Usuario*

Realiza la generación y mantenimiento de los usuarios del lector de noticias, permitiendo setear las propiedades básicas para el uso de la aplicación, como por ejemplo el servidor de noticias al cual se conecta. También ofrece la posibilidad de importar y exportar archivos newsrc y setear propiedades funcionales para cada usuario.

⇒ *Lector de Noticias*

Desarrollo de las clases componentes para la interfaz e integración funcional de la aplicación, acceso al editor, mantenimiento de la agenda de direcciones, conexión al servidor, etc..

⇒ *Editor*

Implementación de las clases componentes para el funcionamiento y visualización del editor de noticias y mail.

CLIENTE NNTP Y SMTP

Clase: ClienteNNTP

Esta clase representa la implementación del protocolo NNTP, para el intercambio de los mensajes, en este caso la aplicación establece una comunicación con el servidor de noticias elegido. La comunicación se establece mediante la creación de sockets y se utilizan buffers de memoria para realizar los procesos de entrada y salida. De esta manera cuando se produce la entrada de datos, se lee de una vez toda la información de los grupos de noticias o artículos, y luego la aplicación leerá los datos desde el buffer de entrada. De la misma manera, cuando por ejemplo se publica un artículo, los datos de salida quedan almacenados en el buffer de salida.

Métodos principales:

- `openServer(serverName,119)`
Se conecta por medio de la creación de un socket al servidor indicado en `serverName`, en el puerto 119. A su vez se crean, un flujo de entrada "serverInput" por medio de la clase `BufferedInputStream` y un "serverOutput" por medio de la clase `BufferedOutputStream` y `PrintStream`.
- `readServerResponse()`
Modeliza la interpretación de las respuestas que envía el servidor al último comando enviado desde el cliente. Estas respuestas están formadas por un código de estado, de 3 dígitos y una parte de parámetros. Algunas respuestas contienen parámetros tales como números y nombres. Los parámetros están separados del código y uno de otro por un espacio. Todos los parámetros numéricos son decimales y pueden tener ceros. Todos los parámetros string comienzan después de un espacio y por lo tanto finalizan antes del

siguiente o del par CR-LF al final de la línea. Este método mantiene un vector con los parámetros enviados y retorna el código de estado, de modo que la aplicación pueda interpretar el éxito o falla del comando antes de mostrar los resultados obtenidos.

- `EnviarComandoServer(comando)`

Se encarga de enviar efectivamente el comando al servidor, mediante el buffer de salida y obtener el código de respuesta enviado por el servidor. Si el código de respuesta es menor que 500 indica que el comando no puede ser realizado por alguna razón inusual, en este caso, se desconecta del servidor y vuelve a conectarse mediante el método `penServer`.

- `getGroup(name)`

El parámetro es el nombre del grupo ha ser seleccionado. Modeliza el comando "group" del protocolo. Si el servidor envía el código 211 la respuesta es exitosa y además del código retornará el primero y el último artículo del grupo, y un estimado del número de artículos en dicho grupo. Si el código de retorno es 411 indica un grupo de noticias inválido. Si la respuesta es exitosa se creará un objeto Grupo con la información del grupo de noticias, si la respuesta es fallida, se genera una `LectorException`.

- `getList()`

Corresponde al comando "list" y retorna una lista de grupos de noticias disponibles y su información asociada. La información de cada grupo de noticias es enviada como una línea de texto, indicando el nombre del grupo, el último artículo en el grupo, el primer artículo y una indicación si es posible publicar artículos en dicho grupo o no. Los campos primero y último son siempre numéricos y pueden ser ceros. Si el campo último es menor que el primero, indica que no hay artículos en el grupo.

- `getArticle(int n)`

Corresponde al comando "ARTICLE n", si el artículo corresponde al grupo el servidor enviará el código de estado 220 y luego el header del artículo, una línea en blanco y a continuación el cuerpo del artículo. Este método retornará el cuerpo del artículo mediante un objeto `InputStream`.

- `getHeader(int n)`

Corresponde al comando HEAD. El servidor enviará el código de estado 221 y luego el header del artículo.

- `startPost()`

Es la implementación al comando POST. Si el posting es permitido en servidor enviará el código de estado 340, indicando que el artículo debería ser enviado. Si el código es 440 indica que el posting no está permitido en este grupo. Si el posting es permitido, el artículo debe ser enviado en el formato especificado e incluir todas las líneas de header requeridas. Después que el header y el cuerpo del artículo han sido completamente enviados por el cliente, el servidor enviará el código de respuesta 240 indicando el éxito o el 441 falla del intento de posting. En caso de ser permitido el posting, este método retorna un objeto `PrintStream` para el envío del artículo.

- `finishPost()`

Finaliza el post, enviando al servidor ".". Este método retorna true si el servidor envió el código 240.

Clases: `Message` y `Transport` de la API `JavaMail`

Message es una clase abstracta que representa un mensaje de mail. `JavaMail` implementa RFC822 y MIME. **Transport** es otra clase abstracta que representa un protocolo de transporte específico.

El mensaje de correo electrónico, es implementado mediante la clase **Message**, que define un conjunto de atributos que representan la información de la dirección remitente, destinatario y asunto, y la estructura del contenido del mensaje. Los mensajes pueden contener un único contenido, o múltiples contenidos. En cualquiera de los casos el contenido es manejado por un objeto **DataHandler**.

Los mensajes de contenido múltiple, tienen múltiples partes, cada una de estas es, en sí misma, un mensaje que tiene su propio conjunto de atributos y contenido. El contenido de un mensaje de múltiples partes es un objeto **Multipart** que contiene objetos **BodyPart** que representan cada parte individual. En este caso el mensaje tiene un atributo que indica que tiene múltiples partes y el contenido es una referencia a un objeto **Multipart**, el cual puede contener simplemente un **DataHandler** o bien otro objeto **Multipart**.

Como la clase **Message** es abstracta, para crear un mensaje se debe instanciar una subclase. Un mensaje se crea instanciando la clase **MimeMessage** y luego que el mensaje es creado, se

setean los atributos, se inserta el contenido y se llama al método **Transport.send** que se encargará de rutear el mensaje hasta el receptor.

Los atributos especifican la dirección y otros valores necesarios para enviar, rutear, y decodificar el mensaje. Los atributos también especifican la estructura del mensaje y el tipo del contenido.

Cuando el mensaje tiene múltiples partes, cada una de las partes esta representada por objetos **MimeBodyPart** que son agregados al objeto **Multipart**.

Un objeto **Session** maneja las opciones de configuración y usuario para interactuar con el servidor. Para inicializar un objeto sesión es necesario definir un objeto **Properties** que contenga los valores por default correspondientes a la configuración. Un objeto sesión se utiliza para recuperar un objeto **Transport** basado en las propiedades cargadas para la sesión.

Para crear un mensaje MIME se instancia un objeto **MimeMessage** pasándole como parámetro la sesión actual.

La clase **Message**, provee un conjunto de métodos para especificar los atributos estándares comunes a todos los mensajes. La clase **MimeMessage** provee métodos adicionales que setean atributos MIME. La interface **Part** especifica los siguientes métodos para setear headers especiales:

- `setHeader(String name, String value);`
- `setFrom(Address addr):` especifica explícitamente el emisor como un objeto `Address`
- `setFrom:` recuperará el nombre del emisor del sistema local.

Para agregar contenido a un mensaje, una vez que el contenido está creado, se instancia un objeto **DataHandler**, se agrega el contenido en este último y se coloca dentro del `Message`.

Generalmente se agrega el contenido llamando a **setDataHandler(DataHandler)** del `Message`.

El **DataHandler** es un objeto que encapsula datos, los datos son pasados como parámetros al constructor, ya sea como un **DataSource** (un stream conectado a los datos) o como un objeto Java

Alternativamente, **setContent** implementa una técnica más simple que toma un objeto, y su tipo MIME.

Para enviar el mensaje se llama al método **send()** del objeto **Transport** que se obtiene de la sesión.

Para crear un mensaje Multipart MIME se debe instanciar un objeto **MimeMultipart**, crear un **MimeBodyParts** para cada una de las partes, setear el contenido para los `BodyPart`, como en un mensaje común y finalmente agregar el objeto `Multipart` al mensaje.

GRUPO DE NOTICIAS

Clase: Newsgroup

Esta clase modeliza los grupos de noticias dentro de Usenet. La clase especifica un conjunto de atributos que definen al grupo, como ser el nombre del grupo, el primer y último artículo disponible en él y además si ha sido recuperado y si el usuario está o no suscrito al mismo. Esta información se refiere al grupo de noticias propiamente dicho y de los artículos que se encuentran en él, solo mantiene referencias del primero y último, como así también los que quedan sin leer. Esta clase permite identificar aquellos grupos a los cuales cada usuario se encuentra suscrito, y cuales ha visitado en cada sesión de trabajo, para esto, utiliza la clase auxiliar Grupo, que identifica al grupo de noticias y además mantiene el estado de acuerdo al usuario. Cuando se recupera la información del grupo de noticias, es grabada en una base de datos, de modo de poder disminuir el tiempo de acceso a los grupos recuperados, como así también el tiempo de conexión y de esta forma permitir un acceso al servidor de noticias en modo offline. La información de los grupos de noticias se mantiene de una sesión a otra, de forma que no sea necesario recuperar los grupos de noticias disponibles en el servidor en cada sesión.

Métodos principales:

- Newsgroup(Grupo gr, boolean suscrito, boolean recuperado)
Crea un nuevo grupo de noticias, indicando que ha sido recuperado y si el usuario está o no suscrito.
- firstArticle()
Devuelve el primer artículo disponible en el grupo.
- lastArticle()
Retorna el último artículo disponible en el grupo.
- marcarLeido(int numeroArtículo)

Setea el artículo en estado leído y actualiza la información en la base de datos.

- `marcarRecuperado()`

Coloca el grupo en estado de recuperado y actualiza la información en la base de datos.

- `marcarSuscripto()`

Setea el grupo en estado de suscrito y actualiza la información en la base de datos.

- `name()`

Retorna el nombre del grupo de noticias.

- `leerArchivo(String usuario, boolean soloSuscriptos)`

Devuelve una lista de los grupos de noticias que el usuario ha recuperado, estos grupos podrán ser todos los que se encuentran en la base de datos o solamente los suscritos.

- `releerGrupo(ClienteNNTP news)`

Actualiza la información del grupo almacenado en la base, volviendo a solicitar al servidor los datos correspondientes.

- `newsString()`

Utiliza la clase `JCString` para crear un objeto en el cual poder mezclar imágenes y texto para poder visualizar el panel de grupos mostrando un ícono relacionado en el caso que el usuario este suscrito al grupo.

ARTÍCULOS

Para el tratamiento de los artículos de Usenet, se definen un conjunto de atributos y el contenido del artículo. Los atributos especifican la información del header del artículo y una referencia a un archivo externo, mantiene el cuerpo. Para facilitar el tratamiento de los artículos en los distintos paneles y proveer la facilidad de evitar recuperar artículos no deseados por el usuario, se utilizan dos clases para modelizar un artículo. Cuando el usuario recupera artículos del servidor, estos son representados por objetos de estas clases y se guardan en la base de

datos para su posterior uso. La relación entre artículos esta representada con una estructura tipo árbol, de modo que cada uno tengan una referencia a sus artículos respuestas, como así también mantiene una referencia de aquellos que se encuentran al mismo nivel. Tiene atributos que definen el encabezamiento del artículo, y seteos si el artículo ya fue leído o no.

Clase: HeaderArtículo

Métodos principales:

- HeaderArtículo()
Define un conjunto de atributos, que representan los datos del encabezado del artículo. Cuando el header del artículo es recuperado, el servidor envía una serie de líneas de caracteres terminadas con un CR-LF que representen headers y estas continúan hasta que llega una línea en blanco. Cada línea contiene el nombre del header, seguido de dos puntos (:) y luego el contenido del header. Este método lee carácter a carácter desde el buffer de entrada e identifica los que se encuentran antes y después de (:), con esta información genera dos arreglos, uno que mantiene todos los headers y otro con los contenidos. En el caso que la información del header sea recuperada de la base datos, simplemente se lee cada uno de los contenidos de los headers correspondientes y se crean los arreglos.
- findValue(String clave)
Retorna el contenido del header especificado en 'clave'.
- add(String clave, String valor)
Agrega el header y el contenido del mismo a los arreglos internos correspondientes.

Clase: Articulo

Métodos principales:

- Artículo(String asunto, String autor, String num, String id, String fecha, String líneas, String referencias, boolean leído, String grupo)

Crea un artículo nuevo, con la información del asunto, autor, número del artículo dentro del grupo, identificador del artículo, cantidad de líneas, fecha de creación, referencias que tendrán los identificadores de los artículos a los cuales está respondiendo en el caso de ser una respuesta, si está leído, y el grupo al que pertenece el artículo.

El atributo asunto, puede contener una repetición de Re: o re: cuando el artículo es una respuesta, para evitar esa sucesión, estos son eliminados antes de crear el objeto, como así también los espacios en blanco que pueda contener.

El atributo referencias: es un string que contiene todos los identificadores de los artículos a los cuales es respuesta, se crea un vector con las referencias.

Cada artículo, tiene dos atributos especiales:

resp: que es un puntero a un artículo que es respuesta del actual.

next: que es un puntero a un artículo que se encuentra al mismo nivel que el actual.

- insert(Artículo raíz, boolean toplevel)

Este método se utiliza para crear un árbol con todos los artículos del grupo, manteniendo la relación entre artículos originales y respuestas. Mediante el atributo "referencias" verifica que un artículo sea respuesta de los artículos ya recibidos en forma recursiva los va referenciando como respuesta de un artículo o no.

El primer artículo que se recupera del servidor se considera inicialmente la raíz del árbol, y este artículo todavía no tiene ni respuestas ni artículos al mismo nivel. Al recuperar otro artículo, existen tres posibilidades:

- 1) que el artículo nuevo, tenga en su vector de referencias a la raíz, en cuyo caso, el artículo actual es una respuesta de la raíz.
- 2) que la raíz tenga en su vector de referencias el identificador del artículo nuevo, en cuyo caso, el artículo nuevo se transforma en la nueva raíz, y la anterior raíz pasaría a ser una respuesta del nuevo, luego hay que evaluar el subarbol descendiente de la vieja raíz, y colocarlos ya sea como respuesta del nuevo o al mismo nivel.
- 3) que el artículo nuevo se encuentre al mismo nivel que el actual.

- toVector(Grupo ng)

Retorna un vector que contiene la información formateada con imágenes, colores que representan el estado actual del artículo, en la sesión de trabajo, es utilizado para la visualización de los artículos en el Panel de artículos.

- `grabarBDD(String usuario)`
Graba en la base de datos cada uno de los atributos del artículo actual.
- `leerArtículo(ClienteNntp news)`
Recupera el cuerpo del artículo del servidor de noticias. Crea un objeto `HeaderArtículo`, y luego lee el cuerpo del artículo línea a línea. Marca el artículo como leído y graba el cuerpo del artículo en un archivo externo, para poder ser recuperado posteriormente.

USUARIO

Clase: `PanelUsuario`

Esta clase extiende de la clase **`JPanel`** y permite la generación y mantenimiento de los usuarios del lector de noticias. Las clases abstractas más importantes utilizadas en la implementación son **`JTabbedPane`**, **`JPasswordField`** y **`JOptionPane`**.

Métodos principales:

- `PanelUsuario()`
Despliega la ventana desde la cual se podrá entrar a la aplicación con un usuario existente o elegir la opción de crear un nuevo usuario.
- `crearPanelUsuario()`
Permite ingresar los datos de un nuevo usuario o modificar la información de un usuario existente. Estos datos son el nombre, la clave, la dirección de mail, el servidor smtp y el servidor nntp.
- `crearPanelImportar()`
Visualiza la ventana a través de la cual se obtiene la información del archivo newsrc que el usuario desea importar.
- `crearPanelExportar()`
Despliega la ventana a través de la cual se obtiene la información del nombre del archivo que se generará en la exportación.

- `crearPanelOpciones()`
Visualiza la ventana a través de la cual el usuario podrá setear las propiedades de incluir texto original en respuestas, marcar todos los artículos como leídos y eliminar headers no disponibles.
- `crearUsuario()`
Realiza un llamado a la clase **Usuario** para hacer efectiva la creación del nuevo usuario del lector de noticias.

Clase: Usuario

Métodos principales:

- `grabarUsuario(String nombre, String clave, String mail, String smtp, String sntp)`
Almacena en la base de datos la información de un nuevo usuario o la modificación de los datos del mismo.
- `validaClave(String c)`
Realiza la validación de la clave ingresada por el usuario que utilizará el lector de noticias, retorna true si es correcta y false en caso contrario.
- `leerNewsrsrcFile(String name, String usuario)`
Almacena en la base de datos la información del archivo newsrsrc que el usuario desea importar.
- `crearNewsrsrcFile(String name, String usuario)`
Obtiene de la base de datos la información del archivo newsrsrc que el usuario desea exportar.

LECTOR DE NOTICIAS

Clase: LectorNoticias

Interrelaciona las principales clases y facilita la obtención de los datos para ser utilizado desde la interface.

Métodos principales:

- ConectarServidor()

Inicia una sesión de trabajo, estableciendo una conexión al servidor de noticias especificado en las propiedades del usuario. Para realizar la conexión crea una instancia de la clase ClienteNNTP.

- recuperarGrupo(Newsgroup ng)

Este método se utiliza para recuperar los headers de los artículos pertenecientes al grupo especificado en el parámetro "ng". Se utiliza el método setearGrupo de la clase ClienteNNTP para colocar como activo el grupo seleccionado y se van recuperando desde el primer artículo del grupo, ng.firstArticle(), hasta el último, ng.lastArticle(), cada header se solicita al servidor a través del método getHeader de la clase ClienteNNTP y con el stream obtenido, se crea un objeto de la clase HeaderArtículo. Los headers recuperados son almacenados en la base de datos y luego son tomados desde allí y se genera un árbol en donde se refleja la relación que existe entre los artículos del grupo, este árbol será tomado posteriormente desde la interface para su visualización.

- RecuperarGrupo (Newsgroups ng, int cant)

Funciona en forma similar al anterior, pero solo se recuperan la cantidad de headers indicados en el parámetro "cant", comenzando desde el primero en el grupo "ng". Este método se utiliza cuando se quiere solamente una muestra de los headers del grupo, por ejemplo en el caso que se quiera evaluar si el grupo es del interés del usuario o no, antes de recuperar todos los headers disponibles.

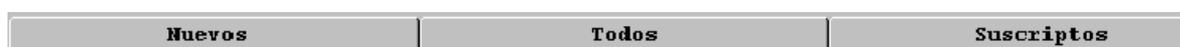
- recuperarGrupoBDD(String usuario, Newsgroup ng)

Este método recupera de la base de datos todos los headers del grupo indicado en el parámetro "cant" por el usuario especificado en "usuario".

Clase: LectorFrame

Esta clase modeliza la interface del usuario. La interface esta compuesta por una barra de herramientas, una serie de pestañas y un conjunto de paneles que facilitan el acceso a la información de los grupos y de los artículos.

Las pestañas permiten discriminar entre visualizar todos los grupos de noticias disponibles en el servidor, solamente aquellos a los cuales el usuario esta suscrito o los nuevos. Están implementadas con la clase **TabsetControl** que crea un conjunto de pestañas mutuamente excluyentes, que permite mediante el método **getSelectedTab()** determinar que grupos desea visualizar el usuario.



Los grupos de noticias, los headers de los artículos y el contenido de los artículos, están ubicados en un **SplitPanel()**, cada uno de los paneles aparece con una barra de separación movable, de manera que puedan ser redimensionados por el usuario.

Los grupos son mostrados en el panel de grupos, el cual esta implementado con la clase **JList** que permite mostrar una lista desplazable de items que pueden contener imágenes además de texto,



de esta manera se muestra una imagen a la izquierda del nombre de un grupo, para indicar que el usuario esta suscrito al mismo.

El funcionamiento de esta clase es el siguiente:

- ⇒ Si se hace 'click' sobre el nombre de un grupo (o se presiona la barra espaciadora) sin ninguna otra tecla oprimida, el grupo es seleccionado o deseleccionado según si ya había sido seleccionado o no respectivamente.
- ⇒ Si se hace 'click' sobre el nombre de un grupo (o se presiona la barra espaciadora) manteniendo oprimida la tecla SHIFT, la selección se extiende a los siguientes grupos.
- ⇒ Si se hace 'click' sobre el nombre de un grupo (o se presiona la barra espaciadora) manteniendo oprimida la tecla CONTROL, permite seleccionar más de un grupo.
- ⇒ Las teclas de cursor permiten moverse a través de la lista de grupos, y el grupo que actualmente tenga el 'foco' será recuadrado con una línea punteada.
- ⇒ Es posible tipear un carácter para que el primer grupo que comience con ese carácter se haga visible y sea seleccionado.

Cada ítem de la lista es un objeto de la clase **JCString** que permite combinar el uso de imágenes y texto, además de seleccionar múltiples colores y fuentes. El manejo de eventos sobre la lista de grupos está implementado con la interfaz **JCActionListener** y el registro con la lista se realiza a través del método **addActionListener**.

El panel de artículos, visualiza los headers de artículos del grupo de noticias seleccionado y está representado por un **JCMultiColumnList** para visualizar en varias columnas el asunto y el autor, la cantidad de líneas del artículo, la fecha de creación y el estado (si fue recuperado, si está marcado, etc.). Además este tipo de listas permiten al usuario ordenar los headers de los artículos haciendo click sobre la cabecera de las columnas y el ancho de las columnas puede ser modificado moviendo las cabeceras de las mismas.

La información relacionada al header del artículo y al autor es visualizada mediante un **JCOutliner** que permite mostrarlos ordenados jerárquicamente en forma de árbol. Cada ítem en el árbol es llamado nodo y corresponde al header de un artículo dentro del grupo. Hay dos tipos de artículos, los que tienen una o más respuestas y los que no. Cada respuesta está indentada hacia la derecha del artículo original y los artículos con respuestas mantienen un ícono a su izquierda.

Asunto	Estado	Líneas	Fecha
 microsoft.public.es.word			
 Problemas con archivos de ayuda (mbenzriv@scinet.com.mx)		9	28/02/2000
└─ Re: Problemas con archivos de ayuda (amaury@retemail.es)		24	29/02/2000
 Re: Crear Cuadernillos (amaury@retemail.es)		35	28/02/2000
└─ Re: Word va muy lento (amaury@retemail.es)		31	29/02/2000
└─ RE: !! URGENTE (jrios@ssaconcagua.cl)		56	1/03/2000

El usuario puede controlar cuando ver o no las respuestas, es posible expandir o contraer los headers haciendo doble click en el mismo, presionando la barra espaciadora o haciendo click en el ícono mostrado a la izquierda de un header que tiene respuestas. Típear un carácter hace que el primer header que comience con dicho carácter tenga el foco. El manejo de eventos sobre el **JCOutliner** se realiza mediante el uso de la interfaz **JCActionListener**, la cual se registra con el **JCOutliner** por medio del método **addActionListener()**.

El contenido de los artículos seleccionados es mostrado en el panel inferior, en donde se visualiza parte de la información del header del artículo además del cuerpo del mismo.

Clase: Direcciones

Esta clase fue implementada para el manejo de la agenda de direcciones, la cual extiende de la clase **JFrame**.

Se encuentra constituida por una ventana la cual despliega una lista, implementada con la clase **List** donde se mostrarán los alias de las direcciones y dos áreas de texto, desarrolladas con la clase **TextArea**, una para la dirección de mail y otra para comentarios.

Al desplegarse la ventana se muestran todas las direcciones almacenadas hasta el momento, accediendo a través del driver jdbc:odbc a la base de datos Lector y dentro de esta a la Tabla Direcciones.

Se encuentra disponible en esta pantalla una barra de herramientas implementada con la clase **JToolBar**, en ella encontramos todas las acciones posibles para el mantenimiento de las direcciones:

- ⇒ Nueva: se realiza un llamado al método **initDNNueva()**. Este método levanta una instancia dDNueva de la clase **JDialog**, a través del cual se pueden ingresar los datos de una nueva dirección, dando la posibilidad de chequear si es correcta presionando el botón de check, quien realiza una reacción al evento **actionPerformed**. Una vez aceptada la nueva dirección se hace nuevamente una reacción al evento **actionPerformed** para hacer efectivo el ingreso de la información a la base de datos.
- ⇒ Borrar: se realiza una reacción al evento **actionPerformed** para asentar la eliminación en la base de datos.
- ⇒ Guardar: se realiza una reacción al evento **actionPerformed** para asentar cualquier modificación de la dirección o comentarios en la base de datos.
- ⇒ Renombrar: se realiza un llamado al método **initDNRenom()**. Este método levanta una instancia dDRenom de la clase **JDialog**, a través del cual se permite especificar un nuevo nombre para la dirección, quien realiza una reacción al evento **actionPerformed** para asentar la modificación en la base de datos.
- ⇒ Chequear: se realiza una reacción al evento **actionPerformed** para verificar si la dirección ingresada tiene un formato válido.

EDITOR

Clase: EditorFrame

Sobre esta clase se desarrolla el editor la cual extiende de la clase **JPanel**.

Esta clase implementa las tres secciones principales del editor: encabezados, barras de herramientas y contenido.

Métodos Principales:

- createEditor()
Crea un **JTextComponent**, la clase base para los componentes de texto en Swing, con un modelo de documento específico.
Este documento, embebido en el editor, es creado con la clase **DefaultStyledDocument** la cual provee un conjunto de estilos por defecto.
- getEditor()
Devuelve el editor contenido en el JPanell.

Encabezados

Esta parte del editor está compuesta por dos etiquetas y sus correspondientes campos de textos para identificar el grupo de noticias al cual se le envía el mensaje y detallar el asunto del mismo; en este caso se ha utilizado respectivamente instancias de las clases **JLabel** y **JTextField**.

Para el ingreso de destinatarios se utilizan dos botones, implementados con la clase **JButton** y sus correspondientes campos de texto donde se especificaran las direcciones de mail, estos últimos implementados con la clase **JTextField**.

Presionando los botones mencionados, se realiza una llamada al evento **actionPerformed** el cual despliega una cuadro de diálogo instanciando la clase **JDialog**, el mismo permite seleccionar direcciones y actualizar automáticamente los campos de textos correspondientes. Estas direcciones son obtenidas de la base de datos Lector, específicamente de la Tabla Direcciones a través del driver jdbc:odbc.

El cuadro de diálogo mencionado está compuesto por tres listas que instancian a la clase **List**, una para mostrar y seleccionar las direcciones y las otras dos mantienen las direcciones de los destinatarios mientras se van agregando. También se visualizan tres botones, instancias de la clase **JButton**, los tres producen una reacción al evento **actionPerformed**, los dos primeros produciendo la actualización de los destinatarios según corresponda que sean principales o a

los que se les copia el mensaje y el otro para aceptar las direcciones produciendo la actualización de los campos de texto y cerrando el cuadro de diálogo.

- `initDDirec()`
Inicializa el cuadro de diálogo que se despliega al seleccionar una dirección con los objetos mencionados en el párrafo anterior.
- `MostrarDirec()`
Realiza la conexión a la base de datos para mostrar las direcciones existentes y el tratamiento de la selección de una o varias de ellas para completar el encabezado del mensaje.
- `SetAsunto(String asunto)`
Rellena automáticamente el campo de texto "Asunto" cuando se desea enviar una respuesta a un artículo. Este método es llamado desde la clase `LectorFrame`.
- `SetAutor (String autor)`
Rellena automáticamente el campo de texto "Mail a" cuando se desea enviar una respuesta al autor de un artículo. Este método es llamado desde la clase `LectorFrame`.

Barra de Herramientas

En esta sección se implementa las barras de herramientas horizontales y verticales del editor. En las barras horizontales, se realiza el manejo de archivos, el formato del texto, la alineación de los párrafos, el manejo del portapapeles, buscar patrones de texto e insertar attachment. La barra de herramientas vertical está preparada para el envío de los mensajes.

- `createToolbar()`
Este método crea y retorna una barra de herramientas horizontal, obteniendo sus componentes del archivo de localización `EditorFrame.properties` a través del método **`getResourceString`**. Para ello también se utiliza el método `createTool`.
- `createTool(String key)`
Manda a crear los botones para la barra de herramientas a través del método `createToolbarButton`.
- `createToolbarButton(String key)`

Este método crea un botón que será componente de la barra de herramientas. Para ello, obtiene a través del método `getResource` la url donde estará la imagen del botón y a través del método `getResourceString` cual es la acción del botón y el hint del mismo.

- `getToolbar()`
Devuelve la barra de herramientas especificada.
- `getFrame()`
Devuelve el frame que será utilizado por los diálogos para la elección de archivos utilizados por las acciones de abrir, guardar y attachment.
- `getActions()`
Retorna la lista de acciones soportadas por la clase **JtextComponent** por defecto, más las acciones definidas localmente para este editor.
- `Action getAction(String cmd)`
Devuelve la Acción correspondiente al string 'cmd'.
- `getResourceString(String nm)`
Devuelve el string correspondiente a la cadena 'nm' buscando en el archivo de localización `EditorFrame.properties`.
- `getResource(String key)`
Devuelve la URL correspondiente al string 'nm' buscando en el archivo de localización `EditorFrame.properties`.
- `tokenize(String input)`
Toma el string dado y lo corta en una serie de string con espacios en blanco. Este método es utilizado por el método `createToolbar`, obtiene un string con los componentes de la barra de herramientas.

Contenido

Esta sección esta implementada a través de la clase **JtextPane**, crea un área de texto la cual tiene embebido un documento creado a través de la clase **DefaultStyledDocument**.

Además de poder aplicar todos los formatos de texto utilizando las barras de herramientas como se explicó anteriormente, se permite hacer scroll de páginas, esta función fue implementada utilizando la clase **JscrollPane**.

- **EnviarMensaje()**
A través de este método se realiza el control si se desea enviar un mail a una persona/s en particular o postear un artículo a todo un grupo de noticias. De acuerdo a la acción elegida se realiza la llamada a los métodos **EnviarMail** o **EnviarNews** según corresponda.
- **EnviarMail()**
Se envía un artículo vía E-Mail. Para esta acción se crean instancias de las clases **Message** y **Transport** de la API **JavaMail**.
- **EnviarNews()**
Hace efectivo el post del artículo. Para esta acción se crea una instancia de la clase **ClienteNNTP**.
- **SetArticulo(String articulo)**
Transfiere el texto de un artículo al editor para realizar una respuesta al mismo. Este método es llamado desde la clase **LectorFrame**.

Clase: OpenAction

Extiende de la clase **AbstractAction** e implementa la acción de abrir un documento almacenado. Se utiliza la clase **FileDialog** para seleccionar la ruta y archivo del documento elegido.

Clase: SaveAction

Extiende de la clase **AbstractAction** e implementa la acción de salvar el documento corriente. Se utiliza la clase **FileDialog** para seleccionar la ruta y archivo en cual será salvado el documento.

Clase: NewAction

Extiende de la clase `AbstractAction` e implementa la acción de crear un documento vacío.

Clase: `Dcolor`

Extiende de la clase `JDialog`. En ella se definen los colores que soporta el editor, y se crean los botones que mostrarán las opciones en el cuadro de diálogo para su posterior selección por parte del usuario. Para cada botón color, se crea una acción a través del método `ForegroundAction` de la clase `StyledEditorKit` y se setea la acción a un comando a través del método `setActionCommand`.

Una vez elegido el color, presionando el botón del mismo, se realiza una llamada al evento `mouseClicked`, produciendo el mismo el cierre del cuadro de diálogo y trasladando la elección al editor para que refleje el cambio de color.

Clase: `Dletras`

Extiende de la clase `JDialog`. En ella se definen los tipos de letras que soporta el editor, y se crean botones que aparecerán en el cuadro de diálogo para su posterior selección por parte del usuario. Para cada botón letra, se crea una acción a través de `StyledEditorKit.FontFamilyAction` y se setea la acción a un comando a través del método `setActionCommand`

Una vez elegido el tipo de letra, presionando el botón del mismo, se realiza una llamada al evento de reacción del mouse, `mouseClicked`, produciendo el mismo el cierre del cuadro de diálogo y trasladando la elección al editor para que refleje el cambio de tipo de letra.

Clase: `Dtamano`

Extiende de la clase `JDialog`. En ella se definen los tamaños de letra que soporta el editor, y se crean los botones que aparecerán en el cuadro de diálogo para su posterior selección por parte del usuario. Para cada botón tamaño, se crea una acción a través de `StyledEditorKit` y se setea la acción a un comando a través del método `setActionCommand`.

Una vez elegido el tamaño de letra, presionando el botón del mismo, se realiza una llamada al evento de reacción del mouse, `mouseClicked`, produciendo el mismo el cierre del cuadro de diálogo y trasladando la elección al editor para que refleje el cambio de tamaño de letra.

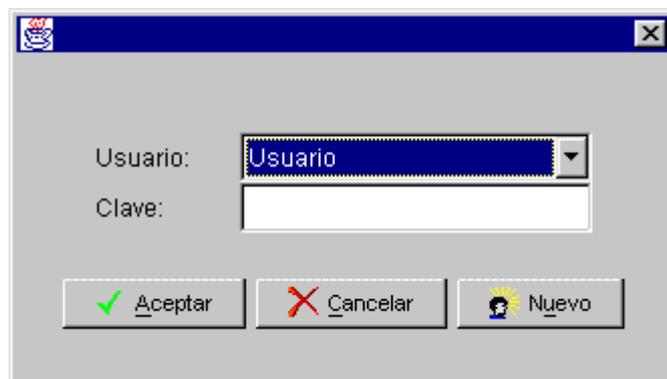
FUNCIONALIDAD DE LA APLICACIÓN

CARACTERÍSTICAS

- ⇒ Permite trabajar en forma offline y online.
- ⇒ Utiliza threading multi nivel, indentando respuestas para mensajes bajo el artículo inicial, así se puede continuar una discusión sin tener que navegar buscando las respuestas.
- ⇒ Tiene incorporado un editor propio para la construcción y lectura de artículos.
- ⇒ Da la posibilidad de enviar mensajes de mail.
- ⇒ Permite el envío de attachment.
- ⇒ Mantiene libreta de direcciones para identificar aquellas direcciones de mails más frecuentemente usadas.
- ⇒ Permite realizar búsquedas por el nombre del grupo o artículos por subject o autor.
- ⇒ Es posible leer y escribir archivos newsrc.
- ⇒ Da la posibilidad de realizar filtros en el momento de seleccionar los grupos de noticias, se puede trabajar con todos los grupos disponibles en el servidor, solamente con los suscritos o los grupos nuevos.
- ⇒ Permite personalizar la cantidad de headers que se recuperan de un grupo.
- ⇒ Mantiene una interface amigable con el usuario, tal que resulta sencillo distinguir entre los artículos que han sido recuperados, los que están aun sin leer y aquellos que son nuevos.
- ⇒ Provee mensajes de error.

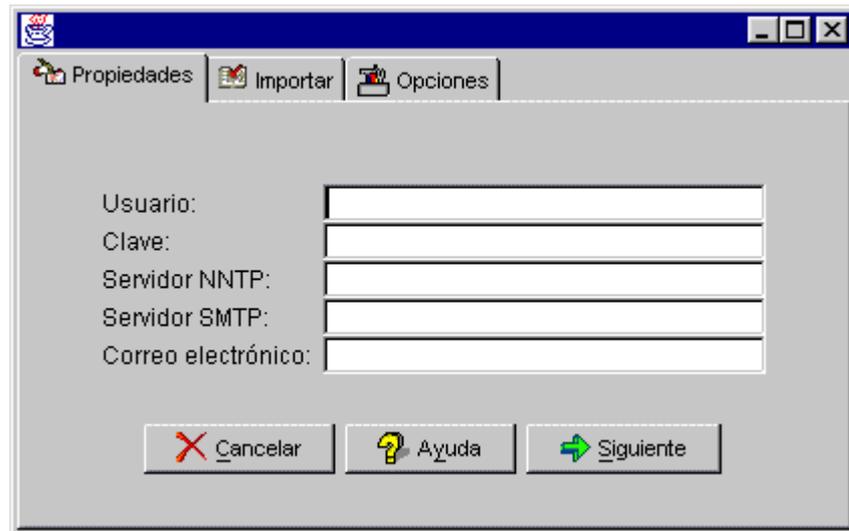
COMIENZO

Al iniciar la aplicación se despliega una pantalla de presentación desde la cual se permite realizar dos acciones, el acceso a una nueva sesión de un usuario existente o la creación de un nuevo usuario del Lector de Noticias.



Para ingresar a la aplicación el usuario debe indicar su nombre, su clave y presionar el botón , de modo de poder recuperar toda la información relacionada con él; si la clave ingresada no es la correcta se emite un mensaje de error y se vuelve a la pantalla anterior dando la posibilidad de ingresarla nuevamente.

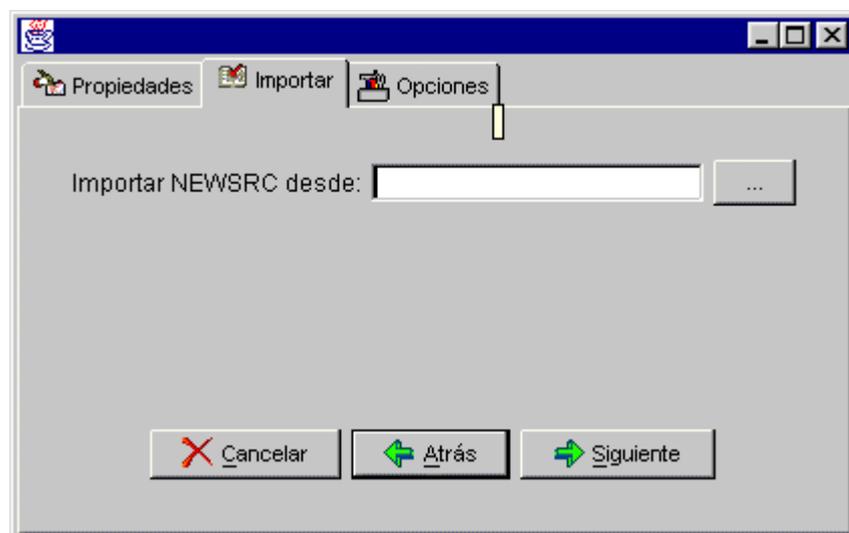
Para crear un nuevo usuario, se elige la opción  y se procede a indicar el nombre del usuario, su clave de acceso, servidor de noticias (Servidor NNTP), el servidor de correo electrónico (Servidor SMTP), y la dirección de correo electrónico.



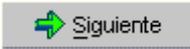
Una vez ingresada la información principal que identifica al nuevo usuario, presionando el botón



, se pasa a la ventana que permite importar un archivo newsrc. Un archivo newsrc es un archivo de texto que contiene los nombres de los grupos de noticias que hayan sido utilizados en otra máquina o accedidos por el usuario a través de otro lector de noticias, junto con información de cada uno de ellos, tales como a cuales se está suscrito y que artículos



ya han sido leídos en cada grupo.

Una vez que se completa la operación, si es de interés del usuario dado que es una acción opcional, presionando nuevamente el botón , se despliega la pantalla que da la posibilidad de setear propiedades funcionales propias del usuario para el uso del lector, como son: incluir texto original en respuestas, marcar todos los artículos como leídos al salir, eliminar headers no disponibles al salir y cantidad de artículos de muestra al recuperar un grupo.

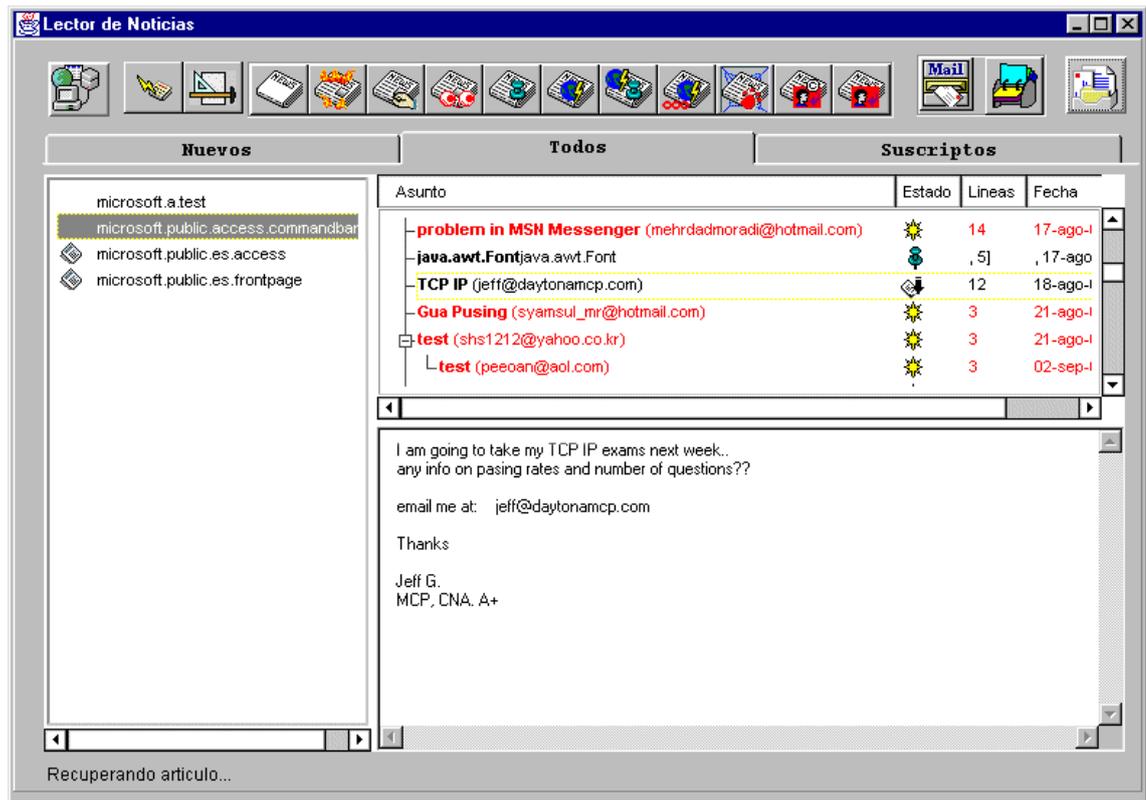


Durante todo el proceso de creación del usuario, en cualquiera de las pantallas, presionando el botón  se vuelve a la pantalla anterior y oprimiendo el botón  se cancela la operación de creación.

Para completar la configuración personal del nuevo usuario se presiona el botón . En el caso que no se haya importado un archivo newsrc, la aplicación solicitará confirmación para conectarse al servidor de noticias especificado y recuperar una lista de grupos. Esta tarea puede llevar unos minutos si el servidor indicado tiene muchos grupos de noticias.

INTERFACE

Cuando esta completa la recuperación de los grupos disponibles, se mostrará la pantalla principal de modo de comenzar a interactuar con los grupos.



La pantalla básicamente esta compuesta por las siguientes secciones:

- ⇒ Barra de herramientas: permite acceder a las operaciones disponibles, tanto para los grupos de noticias como para los artículos, también permite acceder al editor de mensajes, a la libreta de direcciones, y establecer la conexión al servidor.
- ⇒ Las pestañas permiten decidir si se quieren visualizar todos los grupos de noticias disponibles en el servidor, los que hayan sido creados desde la última vez que se realizó una conexión, o bien aquellos en los cuales el usuario está interesado y a los cuales está suscrito.
- ⇒ El panel esta particionado en tres paneles redimensionables: el de grupos, el panel de headers, y el que muestra el contenido del artículo seleccionado.
- ⇒ Barra de mensajes: muestra los distintos estados de ejecución de la aplicación.

NAVEGANDO

La conexión al servidor indicado en la configuración del usuario, se realiza mediante el botón



, una vez completada la operación, se muestra en la línea de estado el resultado de la

misma, indicando si fue exitosa o no, y además si esta permitido o no postear al servidor elegido.

En el panel de grupos se muestran los grupos de noticias disponibles en el servidor, solamente los suscritos o los nuevos. Los grupos son visualizados indicando el nombre y la cantidad de artículos que existen en él. Cuando un usuario realiza una suscripción por medio del botón , se muestra el icono  a la izquierda del nombre del grupo.

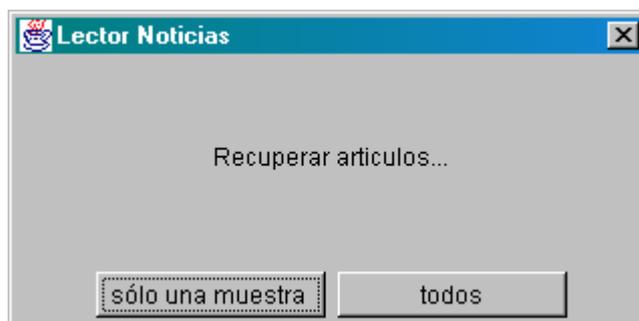
Mediante la reacción del ratón y la utilización de teclas específicas, se permite seleccionar y cambiar en cualquier momento de grupo, artículo, plegar y desplegar threads.

SELECCIONANDO Y LEYENDO ARTÍCULOS

Es posible recuperar los headers de uno o más grupos de noticias a la vez, para esto es necesario seleccionarlos haciendo "click" sobre el grupo. Si se desea seleccionar varios, se debe mantener presionada la tecla "Ctrl" o "Shift" según se encuentren alternados o consecutivos en la lista de grupos respectivamente.

Una vez que han sido seleccionados, se debe presionar el botón  y se obtendrán los headers de los grupos marcados, mostrándose en el panel de headers los correspondientes al primero de los grupos seleccionados. Para poder recuperar los headers es necesario estar conectado al servidor, si el usuario no se ha conectado en el momento de realizar la acción, la aplicación establecerá la conexión automáticamente.

Antes de comenzar a recuperar los mismos, la aplicación solicita al usuario que confirme si desea recuperar solo una muestra de los headers o bien todos los que se encuentren disponibles.



La cantidad de artículos que forman las muestras puede ser indicada en la ventana de opciones accediendo a través del botón . Mientras se realiza la recuperación, en la barra de mensajes se visualiza una indicación de la cantidad de headers que están siendo recuperados y a que grupo de noticias pertenecen.

Estos headers recuperados son almacenados en la base de datos de la aplicación, de modo, que el usuario puede desconectarse, y posteriormente decidir de cuales artículos desea recuperar el contenido.

Para recuperar el cuerpo de un artículo para su lectura, se debe seleccionar el mismo del panel de headers y luego presionar el botón , el artículo será mostrado en el panel inferior.

Los headers de los artículos recuperados son mostrados en color negro y junto a éste se visualiza el icono  que indica que el artículo ya se encuentra almacenado en la base de datos local, de modo que se pueda visualizar en todo momento sin necesidad de conectarse nuevamente.

Cuando se está interesado en un artículo y las respuestas que este contenga, es posible indicarle a la aplicación que se quiere seguir el "thread" de ese artículo marcándolo con el botón , de esta forma la siguiente vez que se recuperen artículos, se obtendrá el artículo marcado y todas sus respuestas.

Inicialmente cuando los headers son recuperados, estos son considerados sin leer y visualizados en color rojo hasta tanto el cuerpo del artículo sea obtenido, en cuyo caso se visualizaran en negro. Debido a que generalmente el usuario no lee todos los headers recuperados, la aplicación mantiene el máximo número de artículos bajados para cada grupo de noticias, de esta manera entre una sesión de trabajo y la siguiente, cuando el usuario vuelve a recuperar headers de un grupo ya visitado la sesión anterior, es posible identificar cuales son los headers de los artículos que han sido posteados al grupo desde la última vez que se conectó, estos serán mostrados en color rojo y junto el icono  indicando que es un header nuevo; los headers que no hayan sido recuperados la última sesión y que aun están disponibles en el servidor, son mostrados en color rojo y el icono  indica que el cuerpo del artículo puede ser recuperado; en caso que el artículo sea eliminado del servidor antes que el usuario pueda recuperar el contenido de dicho artículo, el header del mismo será visualizado en color gris junto con el icono .

Los headers de artículos cuyo contenido no pueda ser recuperado, ya que han sido eliminados del servidor, automáticamente serán eliminados de la base de datos local al finalizar la sesión, si el usuario tiene activada esta opción.

Para eliminar uno o más artículos, es necesario seleccionarlos y luego presionar el botón  que marca los artículos como borrados, de ésta manera se visualizarán en color gris junto al icono  y se eliminarán efectivamente al terminar la sesión. Este comportamiento puede ser modificado desde la ventana de opciones, de modo, que los artículos sean eliminados inmediatamente.

MODOS

La aplicación permite trabajar en modo online/offline. La forma de navegar en modo online es la descrita en los párrafos anteriores. Cuando se trabaja en modo offline, es posible ir marcando los artículos y luego recuperarlos todos juntos, en éste caso, se seleccionan y se presiona el botón , junto al header se mostrará el icono  indicando que el contenido del artículo será recuperado en forma posterior a través del botón .

POSTEANDO ARTÍCULOS

Para realizar el post de un artículo se deben seguir los siguientes pasos:

1. Seleccionar el grupo de noticias al cual se quiere postear: un grupo esta seleccionado si esta resaltado en el panel de grupos o si se esta viendo artículos dentro de ese grupo.
2. Seleccionar el artículo (optativo): en caso de querer responder a un artículo, seleccionarlo primero, así el header del nuevo artículo tomará datos del original.
3. Presionar el botón .
4. Completar los datos del header: si se seleccionó un grupo de noticias y un artículo antes de presionar el botón , muchos de los campos del header contienen datos. Cuando se responde a un artículo, es común dentro de la comunidad Usenet, enviar una copia por e-mail al autor del artículo original, para esto presionar el botón **“Enviar a...”** y seleccionar la dirección del autor de la lista de direcciones.

5. Completar los datos del cuerpo: escribir el texto del artículo. Si es una respuesta y según las opciones que haya configurado el usuario, se muestra en el cuadro de texto, el contenido del artículo original.
6. Enviar el artículo: si no hay conexión activa, la aplicación se conecta en forma automática para enviar el artículo.

ENVIANDO E-MAIL

Para enviar una respuesta a un artículo vía mail se deben seguir los siguientes pasos:

1. Seleccionar el artículo: es necesario para poder enviar una respuesta por e-mail al autor del artículo.
2. Presionar el botón .
3. Escribir la respuesta: los campos del header aparecen completos, escribir el cuerpo del artículo.
4. Insertar un archivo (optativo): si se quiere enviar un archivo adicional, presionar el botón  para buscar, seleccionarlo y se mostrara junto al texto un icono , el nombre y el tamaño en bytes del archivo insertado.
5. Enviar el e-mail: si no hay conexión activa, la aplicación se conecta en forma automática para enviar el e-mail.

BÚSQUEDAS POR SUBJECTS O AUTORES

Se puede buscar grupos por el nombre del grupo o artículos por subject o autor.

⇒ Buscar grupos

1. Seleccionar la pestaña "Todos" para visualizar todos los grupos de noticias.
2. Asegurarse que el panel de grupos esta activo: para realizarlo simplemente seleccionar un grupo.
3. Presionar el botón .

4. Ingresar cualquier parte del nombre del grupo, que se quiere buscar, en la ventana de “**Buscar grupo**” . Se mostrará seleccionado el primer grupo que debajo del actualmente seleccionado que contenga este texto en su nombre.
5. Si el grupo no es el que se esta buscando, presionar <Siguiente> para buscar el siguiente grupo que contenga el texto.

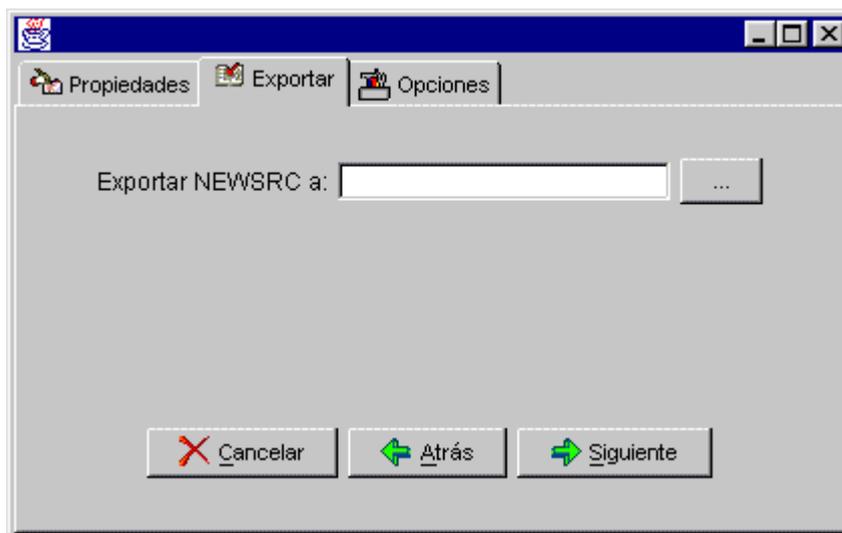
⇒ Buscar artículos

1. Asegurarse que el panel de headers de artículos está activo: para activarlo simplemente seleccionar un header.
2. Presionar el botón .
3. Ingresar cualquier parte del subjetc o nombre del autor. Se mostrará el primer header que contenga el texto entrado.
4. Si el header no es él que se esta buscando, presionar <Siguiente> para buscar el siguiente header que contenga el texto.

PERSONALIZAR

El usuario puede, desde la misma aplicación, cambiar las propiedades definidas por defecto, realizar la exportación de su estado de usuario a un archivo newsrc y cambiar las opciones por defecto,

presionando el botón .



La funcionalidad de las pestañas "Propiedades" y "Opciones" es idéntica a la de creación de un nuevo usuario.

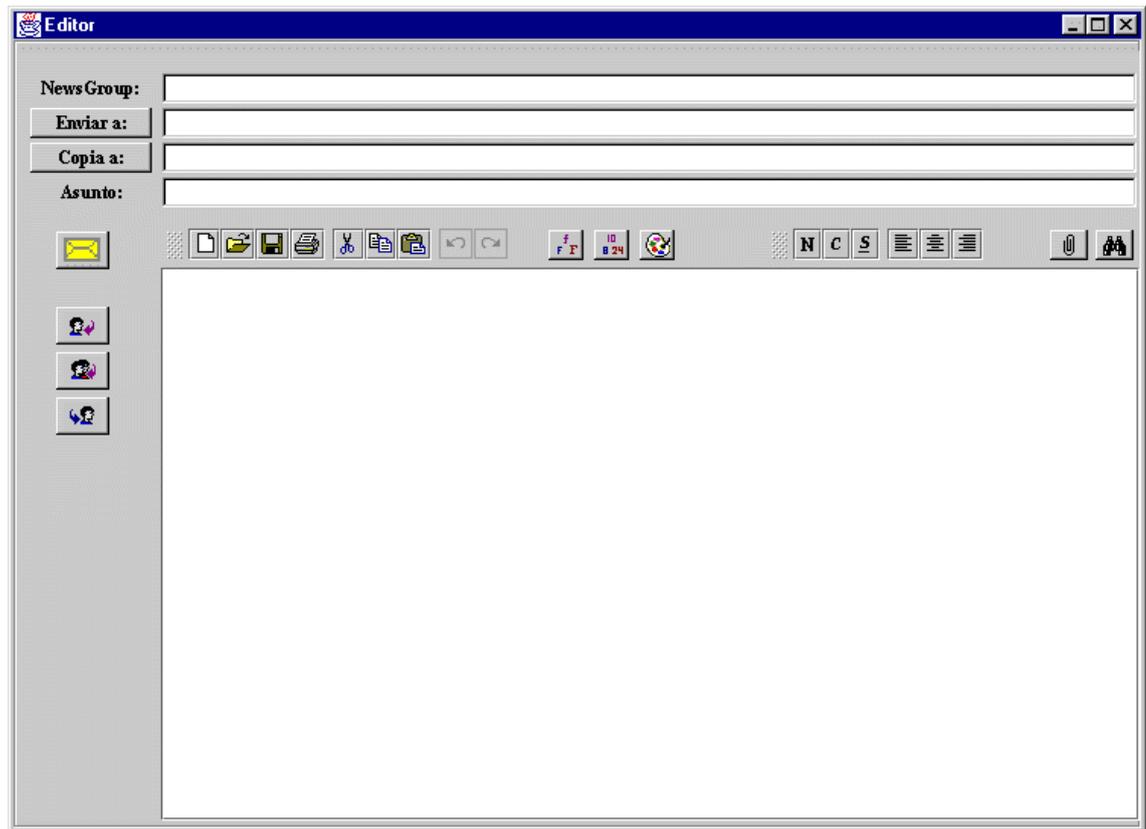
A través de la pestaña "Exportar" es posible volcar toda la información de los grupos de noticias de la sesión actual del usuario al archivo newsrc que se detalla.

UTILIZACIÓN DEL EDITOR

Desde la pantalla Lector de Noticias se puede enviar una respuesta a un artículo recuperado, a todo el grupo de noticias  o al autor del mismo vía mail ; con la primera opción al presionar el botón se desplegará una instancia del editor rellenando automáticamente los campos "Newsgroups", "Asunto" y el "cuerpo" del editor con los datos del artículo, en la segunda opción el comportamiento es idéntico con la salvedad que se rellena el campo "Enviar a" en vez de Newsgroups.

El usuario puede hacer uso del editor presionando el botón  para postear un artículo a un servidor o enviar un mail en forma independiente al Lector de Noticias; en el primer caso se procede de la misma manera que enviar una respuesta detallando en el campo "Newsgroup" el nombre del grupo de noticias y en el segundo caso se deben rellenar los campos "Enviar a", "Copia a", "Asunto" y "cuerpo".

Al seleccionar cualquiera de las opciones descritas se crea una instancia del editor del lector de noticias



El editor está compuesto por tres secciones encabezados, barras de herramientas y cuerpo.

ENCABEZADOS

En esta sección se debe ingresar el nombre del Grupo al cual se va a publicar el artículo "Newsgroup" en caso de enviarlo a todo un grupo de Noticias, la dirección de mail en los campos para él o los destinatarios "Enviar a" y "Copia a" y por último una pequeña referencia del tema tratado "Asunto".

Presionando cualquiera de los botones **Enviar a:** y **Copia a:** se despliega una lista de direcciones para seleccionar las direcciones de mails de los destinatarios almacenadas por el usuario a través del tratamiento de direcciones de la pantalla principal.



Esta pantalla cuenta con tres botones, uno para detallar que el mensaje se va a enviar a una o varias personas en particular , el otro para realizar copias a distintos destinatarios  y el tercero  para aceptar las opciones elegidas, cerrándose automáticamente la ventana de direcciones y rellenándose los campos "Enviar a" y "Copia a" al presionando.

BARRA DE HERRAMIENTAS

El editor posee una serie de barra de herramientas horizontales con las opciones típicas de personalización de texto de un editor.

	Prepara el editor para crear un nuevo e-mail.
	Abre un archivo de texto en el editor.
	Salva a archivo de texto el contenido del editor.
	Imprime el contenido del cuerpo del mensaje.
	Elimina texto seleccionado.
	Copia al portapapeles el texto seleccionado.
	Pega en la posición del cursor el contenido del portapapeles.
	Deshace la última acción.
	Rehace la última acción deshecha.
	Despliega una ventana con los tipos de letras disponibles.
	Despliega una ventana con los tamaños de letras disponibles.
	Despliega una ventana con los colores disponibles.
	Resalta el texto seleccionado.
	Cambia a letra cursiva el texto seleccionado.
	Subraya el texto seleccionado.
	Alinea a izquierda el texto.
	Alinea centrado el texto.
	Alinea a derecha el texto.
	Despliega una ventana con la opción de buscar y reemplazar en el texto.
	Permite insertar un archivo en el documento (attachment).

La barra vertical que se encuentra en el lado izquierdo del editor, se utiliza en el caso de hacer uso de la aplicación como Mail en forma independiente al lector de noticias.

	Responder al autor del mail.
	Responder a todos los integrantes del mail.
	Reenviar el mail.

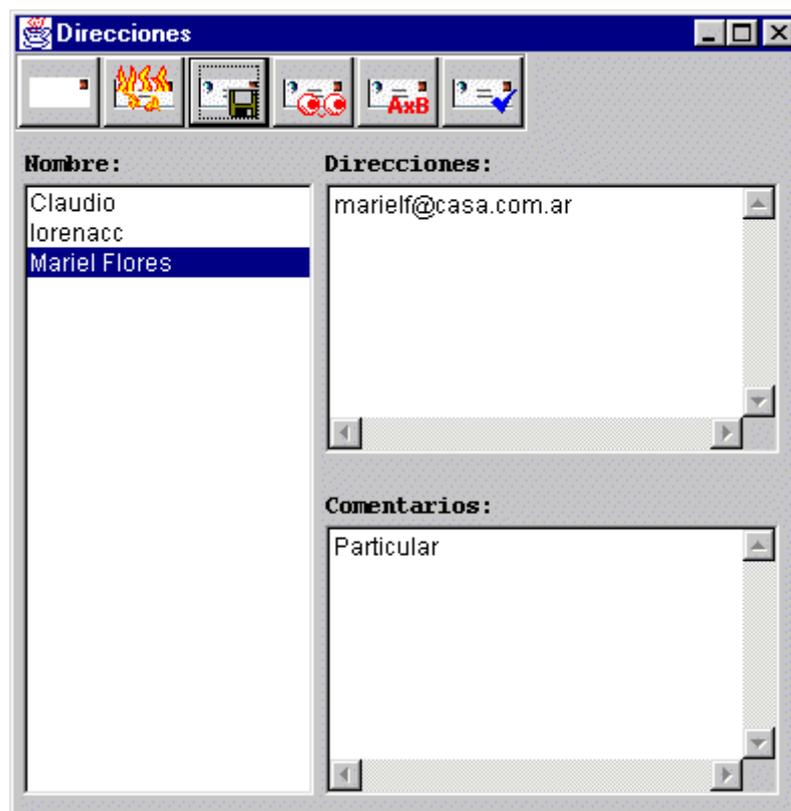
En el nexo entre las barras de herramientas horizontales y la vertical se encuentra el botón de enviar el mensaje, una vez concluido la confección del artículo a publicar o el mail a enviar  se presiona el mismo para hacer efectivo el envío.

CUERPO

La sección cuerpo es el área disponible para el usuario para escribir el artículo o mensaje de mail.

TRATAMIENTO DE DIRECCIONES

Desde la pantalla principal se accede a la ventana de direcciones a través de  y se despliega la ventana:



Esta ventana cuenta con una lista desplegable con los Alias almacenados y dos áreas de texto, una para la dirección y otra para comentarios.

Al desplegarse la ventana se muestran todas las direcciones almacenadas hasta el momento y a medida que se va moviendo a través de la lista en las áreas de texto se va mostrando la dirección y comentario del alias seleccionado, permitiendo realizar una serie de operaciones sobre estas:

⇒ ingresar una nueva , en este caso, se despliega un cuadro de diálogo solicitando al usuario que detalle el alias y la dirección, ofreciéndole además, la posibilidad de chequear que sea correcta a través del botón  y emitiendo un mensaje de validez o invalidez en caso que cumpla o no con el formato de una dirección de mail.



Presionando el botón  se acepta el alta de la nueva dirección cerrando automáticamente la ventana.

⇒ Renombrar un alias existente  verificando que no ingrese duplicaciones.



Presionando el botón  se acepta la modificación cerrando automáticamente la ventana.

⇒ Borrar  una dirección almacenada.

⇒ Habiendo seleccionado un nombre de dirección, se puede modificar la misma permitiendo guardar los cambios realizados  y pudiendo previamente volver a realizar un chequeo de la dirección actualizada .

FINALIZACIÓN

Para terminar de utilizar el lector de noticias se procede como en cualquier aplicación windows y automáticamente se actualiza la información en la base de datos de acuerdo a las opciones personalizadas por cada usuario.

USABILIDAD DEL PRODUCTO LECTOR DE NOTICIAS

La tendencia actual de los proveedores de servicios es dotar a las organizaciones de funcionalidades para la descentralización y agilización de tareas; haciendo uso de Intranet / Internet, se pueden desarrollar las mismas de forma más efectiva, garantizando datos actualizados y exactos, simplificando y automatizando procesos, reduciendo costos, mejorando las comunicaciones internas, etc..

Nuestra aplicación al tener el protocolo de comunicación abierto y extendible puede ser reutilizada para este fin, es posible incorporarle formularios para el ingreso de datos (por medio de Lap Top u organizadores electrónicos) en algún formato y luego los datos ser transferidos por el protocolo de comunicación al sistema correspondiente, logrando de esta manera el comportamiento de una aplicación del tipo "self_service" que permita a los distintos usuarios crear y mantener datos en tiempo real extendiendo la funcionalidad del sistema a toda la organización. Otra posibilidad es la creación de una aplicación del tipo "gestión documental", donde cada usuarios puede aportar y obtener información de una base de datos de conocimientos, accediendo en forma rápida y sencilla.

En resumen, se puede reutilizar el protocolo implementado en nuestra aplicación, extendiéndolo para que actúe como un "middleware" entre aplicaciones e infraestructura de red .

CONCLUSION

Con el desarrollo de nuestro Lector de Noticias, hemos obtenido un producto que cumple con los objetivos básicos de los programas para trabajar dentro de Usenet. El resultado obtenido es una aplicación cliente/servidor de interface amigable e interactiva que integra las funcionalidades de news y mail; implementado en lenguaje Java que constituye, actualmente, la mejor solución para la programación de un cliente, ya que, no solo es un poderoso lenguaje de programación, seguro, multiplataforma e internacional, sino que además esta siendo continuamente extendido para proveer características del lenguaje y librerías que elegantemente solucionan problemas que son difíciles en la programación tradicional, tales como, multithreading, acceso a base de datos, programación de red y programación distribuida. La implementación del lector nos ha permitido obtener un conocimiento más amplio de este atractivo servicio de la Web, como así también ahondar en las bondades del lenguaje Java, utilizado para su desarrollo.

NOCIONES BASICAS

API (Application Programming Interface): conjunto de métodos, funciones, clases y librerías proporcionadas por un lenguaje o un sistema operativo.

Applet: programa de Java que se ejecuta en el contexto de un navegador, y se lanza desde un documento Web, donde se encuentra incrustado.

Archivo de clase: archivo binario con bytecodes de Java. El compilador de Java genera un archivo de clases a partir de un archivo fuente con el código (con la extensión.java) para cada clase de Java.

Archivo de java (Java Archive o JAR): archivo que contiene un conjunto de archivos, como los TAR o los ZIP.

AWT (Abstract Window Toolkit): colección de clases de Java que permiten implementar interfaces gráficas de usuario independientes de la plataforma en la que se ejecute una aplicación.

Bean: componente con software reutilizable, con nombres determinados por convención, que define las propiedades y eventos que extiende. Pueden ser manipulados visualmente mediante una herramienta de construcción.

Bytecodes: código compilado de Java. Son instrucciones transportables que pueden ejecutarse en cualquier Máquina Virtual de Java.

DNS (Domain Name System): base de datos distribuida de Internet que puede convertir direcciones textuales de Internet a sus valores numéricos reales. El DNS organiza como una jerarquía en la que cada nodo se responsabiliza de un subconjunto de direcciones de hosts.

FLAME: guerra de insultos, dentro de la comunidad USENET.

GUI (Graphical User Interface): interfaz gráfica que admite operaciones con el ratón de los sistemas operativos modernos.

HTML (Hypertext Markup Language): lenguaje en el que se escriben los documentos Web. Los applets de Java están incrustados en documentos HTML.

HTTP (Hypertext Transfer Protocol): protocolo de aplicación usado para solicitar, transmitir y recibir documentos Web.

IDE (Integrated Development Environment): programa que ayuda a desarrollar aplicaciones proporcionando un entorno gráfico que combina las herramientas necesarias para escribir código.

IP (Internet Protocol): conjunto de reglas básicas que permiten comunicarse a dos máquinas en una red IP. Se alude también al número IP, que identifica unívocamente a cada máquina de la red.

JDBC (Java Database Connectivity): define un conjunto de clases y métodos de Java para acceder a bases de datos.

JDK (Java Developers Kit): conjunto de herramientas de desarrollo distribuidas por Sun Microsystems. Consiste principalmente en las clases del API del Núcleo de Java, un compilador de Java y el intérprete de Máquina Virtual de Java.

JFC: Clases fundadoras de Java

JGL (Java Generic Library): Librería general de Java

JVM (Java Virtual Machine): Es el programa que interpreta los "bytecodes", las instrucciones Java compiladas.

MIME (Multipurpose Internet Mail Extensions): Protocolo para envío de mensajes multimediales.

MAILING LIST: Listas de direcciones de mail.

NNTP (Network News Transfer Protocol): Protocolo de transferencia de artículos dentro de USENET.

Sitio Web: conjunto de documentos Web que pertenecen a una organización determinada. Un sitio Web puede compartir un sistema servidor con otros sitios Web, o puede extenderse a varios sistemas.

Socket: Uno de los dos extremos de una conexión. Es un elemento de software, que puede ser usado para mantener una comunicación entre dos máquinas.

Swing: API que amplía AWT proporcionando muchos más tipos de componentes GUI.

TCP (Transport Control Protocol): Protocolo de más nivel que el IP. Montado sobre éste permite transmitir paquetes, controlando su efectiva recepción.

Thread: flujo simple de control dentro de un programa, similar a un proceso (o programa de ejecución), pero más fácil de crear y destruir, pues implica una menor gestión de los recursos. Cada thread debe disponer de sus propios recursos, como contador de programa y pila de ejecución, así como del contexto para ejecutarse. Sin embargo, la mayoría de los threads de un programa comparten muchos recursos, como la memoria o los archivos abiertos.

UDP (User Datagram Protocol): Protocolo similar al TCP, pero no tan seguro, pues no controla la recepción de los paquetes, simplemente los envía.

URL (Uniform Resource Locator): Nombre de un recurso en Internet, que sigue un esquema.
Ejemplo: <http://www.javasoft.com/index.html>

WWW (World Wide Web): colección enorme de documentos de hipertexto conectados entre sí en Internet.

BIBLIOGRAFIA

- **Managing Usenet**
Henry Spencer y David Lawrence
O'REILLY
- **The Usenet Handbook**
Mark Harrison
O'REILLY & ASSOCIATES, INC.
- **JAVA Guia de Desarrollo**
Jamie Jaworski
Prentice Hall
- **JAVA 1.2 Al descubierto**
Jamie Jaworski
Prentice Hall
- **Borland Jbuilder**
John Zukowski
Anaya
- **Database Programming with JDBC**
Ashton Hobbs
Sams.net
- **821 Simple Mail Transfer Protocol (SMTP)**
- **977 Network News Transfer Protocol (NNTP)**
- **1036 Standard for Interchange of USENET Messages**
- **Java Mail API Design Specification**
- **JavaBeans Activation Frameworks Specification**

