

Applicazioni comuni per X

29.1	X: configurazione dei clienti	2934
29.1.1	Riga di comando delle applicazioni X	2935
29.1.2	Risorse	2941
29.2	Programmi di servizio tradizionali	2947
29.2.1	Terminale	2947
29.2.2	Clipboard	2949
29.2.3	Caratteri	2953
29.2.4	Informazioni sulle finestre e sul server	2957
29.2.5	Impostazione dello schermo	2962
29.2.6	Programmi di servizio vari	2970
29.3	Gestori di file	2974
29.3.1	XFE	2974
29.3.2	Nautilus	2977
29.3.3	Konqueror	2979
29.3.4	File-roller	2981
29.4	Accesso remoto tradizionale	2982
29.4.1	Putty	2982
29.5	Monitor di vario tipo	2984
29.5.1	Gmemusage	2984
29.5.2	Etherape	2985
29.6	Applicativi comuni per l'ufficio	2986

29.6.1	LibreOffice	2987
29.6.2	Gnumeric	2990
29.6.3	AbiWord	2992
29.6.4	XFig	2993
29.7	Riferimenti	3002
.Xdefaults		2943
.Xresources		2943
/etc/X11/app-defaults/		2943
/etc/X11/Xresources/		2945
bitmap		2968
fig2dev		2998
pstoedit		2998
xbiff		2970
xcalc		2973
xclipboard		2951
xclock		2972
xdpyinfo		2959
xfd		2955
xfe		2974
xferc		2974
xfonstsel		2954
xidle		2970
xkill		2970
xload		2970
xlsfonts		2953
xmem		2970
xrdb		2946
xset		2962
xsetroot		2966
xwininfo		2957
-background		2939
-display		2936
-font		2941
-foreground		2940
-geometry		2937
-title		2940
-xrm		2946

29.1 X: configurazione dei clienti

«

Il funzionamento dei programmi clienti può essere configurato in due modi: con l'uso di opzioni nella riga di comando (cosa comune a tutti i programmi) e attraverso l'impostazione di *risorse*. Alcune opzioni e alcune risorse sono riconosciute dalla maggior parte dei programmi, facilitando il loro utilizzo e rendendo omogeneo il sistema.

29.1.1 Riga di comando delle applicazioni X

Le applicazioni clienti tradizionali permettono di utilizzare, nella riga di comando, una serie di opzioni standardizzate. Si tratta evidentemente di opzioni riferite principalmente all'aspetto del programma, come la dimensione e la colorazione. La tabella 29.1 mostra l'elenco di alcune di queste opzioni.

Tabella 29.1. Alcune delle opzioni comuni ai programmi per X.

Opzione	Descrizione
<code>-foreground</code> <i>colore</i>	Colore di primo piano.
<code>-background</code> <i>colore</i>	Colore dello sfondo.
<code>-reverse</code>	Inverte il colore di primo piano con quello di sfondo.
<code>-bordercolor</code> <i>colore</i>	Colore dei bordi.
<code>-borderwidth</code> <i>dimensione</i>	Dimensione in punti grafici (pixel) dello spessore dei bordi.
<code>-title</code> <i>titolo</i>	Descrizione da porre sulla barra del titolo.
<code>-iconic</code>	Avvia ridotto a icona.
<code>-font</code> <i>carattere</i>	Utilizza il tipo di carattere specificato per visualizzare il testo.
<code>-geometry</code> <i>geometria</i>	Dimensioni e collocazione della finestra.
<code>-display</code> <i>schermo</i>	Coordinate dello schermo su cui deve apparire la finestra.

Nelle sezioni seguenti si analizzano le più importanti.

29.1.1.1 Opzione «-display»

<<

```
-display coordinate_del_display
```

```
-d coordinate_del_display
```

X è fatto per funzionare su sistemi connessi in rete, ognuno dei quali può avere potenzialmente più schermi e può mettere in esecuzione più serverenti grafici: uno per ogni stazione grafica, reale o virtuale che sia. Di conseguenza, per identificare uno schermo di una stazione grafica di un certo elaboratore si utilizza un indirizzo composto nel modo seguente (come già mostrato nel capitolo introduttivo a X).

```
[nodo] : numero_del_servente_grafico [ . numero_dello_schermo ]
```

L'elaboratore può essere identificato attraverso il nome, completo o parziale, oppure con l'indirizzo IP. Quando questa indicazione viene omessa, si intende quello in cui il programma viene messo in esecuzione.

Teoricamente, un elaboratore può mettere in esecuzione contemporanea più di un servente grafico, per pilotare diverse stazioni grafiche. GNU/Linux in particolare, può farlo attraverso delle stazioni grafiche virtuali che si comportano in modo simile a quello delle console virtuali. La numerazione parte da zero, di conseguenza, quando si fa riferimento al primo (e di solito unico) servente grafico a disposizione, si indica semplicemente ':0'.

Teoricamente, un servente grafico può pilotare più di uno schermo

per volta. La numerazione parte da zero, di conseguenza, quando si fa riferimento al primo (e di solito unico) schermo del primo server grafico a disposizione, si indica ‘:0.0’, oppure si omette semplicemente l’indicazione (‘:0’).

La variabile di ambiente **DISPLAY** viene usata per stabilire le coordinate predefinite dello schermo sul quale devono apparire i programmi avviati senza l’indicazione di questa opzione (‘-display’). In situazioni normali, il suo contenuto è ‘:0.0’.

Nell’esempio seguente, si mostra un caso tipico, in cui si avvia un programma in un elaboratore diverso dal proprio e lo si visualizza sul monitor del proprio elaboratore. Tuttavia, perché ciò possa funzionare, occorre abilitare la connessione (questo problema viene analizzato in parte nella sezione [28.5.5](#)).

```
roggen$ telnet dinkel.brot.dg [Invio]
```

...

```
dinkel$ xcalc -display roggen.brot.dg:0 & [Invio]
```

29.1.1.2 Opzione «-geometry»



```
-geometry [dimensioni] [posizione]
```

```
-g [dimensioni] [posizione]
```

Spesso è possibile definire la dimensione e la posizione della finestra iniziale aggiungendo l’opzione ‘-geometry’. Le dimensioni sono espresse secondo la sintassi seguente:

`[=] dimensione_orizzontale × dimensione_verticale`

I valori possono essere espressi in punti grafici (pixel) o in caratteri, a seconda che si tratti di programmi che utilizzano la grafica o meno. Il segno '=' è facoltativo. La posizione viene espressa secondo la sintassi seguente:

`{+|-} distanza_orizzontale {+|-} distanza_verticale`

In pratica si tratta di due valori, espressi in punti grafici, preceduti da un segno: un valore positivo indica una distanza dal margine sinistro o dal margine superiore; un valore negativo, indica una distanza dal margine destro o dal margine inferiore.

Tabella 29.2. Posizione dei quattro angoli dello schermo.

Posizione	Descrizione
+0+0	Angolo superiore sinistro.
-0+0	Angolo superiore destro.
-0-0	Angolo inferiore destro.
+0-0	Angolo inferiore sinistro.

Segue la descrizione di alcuni esempi.

- `$ xterm -geometry +0+0 & [Invio]`

Avvia il programma '**xterm**' sullo sfondo collocando la sua finestra a partire dal punto più alto e più a sinistra possibile della

superficie virtuale attiva.

- `$ xterm -geometry -10-10 & [Invio]`

Avvia il programma `'xterm'` sullo sfondo, collocando l'angolo inferiore destro della sua finestra a 10 punti grafici dal margine destro e dal margine inferiore della superficie virtuale attiva.

- `$ xterm -geometry =80x25+0+0 & [Invio]`

Avvia il programma `'xterm'` sullo sfondo, in una finestra di 80×25 caratteri, collocata a partire dal bordo superiore e sinistro della superficie virtuale attiva.

- `$ xcalc -geometry =500x200+20+10 & [Invio]`

Avvia il programma `'xcalc'` sullo sfondo, in una finestra di 500×200 punti grafici (deformandolo), collocata in modo che l'angolo superiore sinistro della sua finestra si trovi a 20 punti dal margine superiore della superficie virtuale attiva e a 10 punti dal margine sinistro.

29.1.1.3 Opzione «-background»

`-background colore`

`-bg colore`

Questa opzione permette di definire il colore dello sfondo. Il colore viene fornito in forma alfabetica, cioè con l'indicazione del suo nome. I nomi dei colori con le loro corrispondenze RGB sono contenuti nel file `'/etc/X11/rgb.txt'`, così come indicato nel



file di configurazione di X (`/etc/X11/xorg.conf`) nella sezione **Files**.

L'esempio seguente avvia **xcalc** utilizzando il colore denominato **blue** per lo sfondo:

```
$ xcalc -bg blue & [Invio]
```

29.1.1.4 Opzione «`-foreground`»

«

```
-foreground colore
```

```
-fg colore
```

Questa opzione permette di definire il colore di primo piano. Il colore viene fornito in forma alfabetica, cioè con l'indicazione del suo nome. I nomi dei colori con le loro corrispondenze RGB sono contenuti nel file `/etc/X11/rgb.txt`, così come indicato nel file di configurazione di X (`/etc/X11/xorg.conf`) nella sezione **Files**.

L'esempio seguente avvia **xcalc** utilizzando il colore **red** per il primo piano:

```
$ xcalc -fg red & [Invio]
```

29.1.1.5 Opzione «`-title`»

«

```
-title titolo
```


Questa opzione permette di definire un titolo da fare apparire sulla barra superiore della finestra: la barra del titolo.

L'esempio seguente avvia **xcalc** facendo apparire sulla barra del titolo: «calcolatrice tascabile»:

```
$ xcalc -title "calcolatrice tascabile" & [Invio]
```

29.1.1.6 Opzione «-font»



```
-font { nome_del_font | dimensioni_del_font }
```

```
-fn { nome_del_font | dimensioni_del_font }
```

Questa opzione permette di definire il tipo di carattere o la dimensione da utilizzare per le applicazioni che visualizzano testo.

Segue la descrizione di alcuni esempi.

- \$ **xterm** -font 7x14 & [Invio]

Avvia **xterm** utilizzando caratteri di dimensione 7×14.

- \$ **xterm** -font 10x20 & [Invio]

Avvia **xterm** utilizzando caratteri di dimensione 10×20.

- \$ **xterm** -font -adobe-courier-* & [Invio]

Avvia **xterm** utilizzando caratteri **adobe** del tipo **courier**. Il tipo di carattere non viene indicato in modo preciso, per mezzo dell'asterisco finale che aiuta a completarne il nome. In pratica, le altre caratteristiche del tipo di carattere vengono lasciate al loro valore predefinito.

29.1.2 Risorse

<<

Ogni programma cliente può essere configurato attraverso delle *risorse*. Si tratta di qualcosa di paragonabile all'assegnamento di valori a oggetti determinati che rappresentano un elemento o un comportamento particolare di un programma.

Queste risorse sono descritte all'interno di file di configurazione e le relative impostazioni possono essere attivate attraverso l'uso del programma '**xrdb**' (*X resources database*).

I file delle risorse possono contenere dei commenti: il punto esclamativo ('!') viene utilizzato come l'inizio di una riga da ignorare.

29.1.2.1 Nomi delle risorse

<<

Le risorse di ogni programma sono stabilite dal programma stesso e solitamente se ne trova l'elenco nella pagina di manuale relativa. Si tratta normalmente del nome del programma, seguito da altri nomi, separati da un punto, riferiti a elementi di gerarchia inferiore, fino a giungere all'elemento finale. Per esempio, '**XClock.input**'.

Nell'indicazione dei nomi di queste risorse si può utilizzare l'asterisco ('*'), che viene interpretato come avviene nelle shell comuni, così da poter indicare gruppi di risorse in modo semplificato.

Tabella 29.3. Elenco di alcune risorse utilizzate più frequentemente.

Risorsa	Descrizione
*background: <i>colore</i>	Colore dello sfondo.
*foreground: <i>colore</i>	Colore di primo piano.
*borderColor: <i>colore</i>	Colore dei bordi.
*title: <i>titolo</i>	Descrizione da porre sulla barra del titolo.
*iconic: {on off}	Avvia ridotto a icona.
*font <i>carattere</i>	Utilizza il tipo di carattere specificato per visualizzare il testo.
*geometry: <i>geometria</i>	Dimensioni e collocazione della finestra.

29.1.2.2 Configurazione delle applicazioni

Di norma, la directory `/etc/X11/app-defaults/` (oppure `/usr/lib/X11/app-defaults/`) contiene una serie di file, ognuno riferito a un programma particolare, per il quale vengono dichiarati i valori predefiniti delle risorse di sua competenza.

I nomi di questi file sono abbastanza simili a quelli dei programmi a cui si riferiscono. Tuttavia, per sapere esattamente come viene identificato un programma per ciò che riguarda le sue risorse occorre consultare la sua documentazione.

A fianco di questi file, l'utente dovrebbe avere la possibilità di definire un file nella propria directory personale: `~/Xdefaults`. Questo file può essere scritto sfruttando le stesse tecniche di precompila-

zione dei linguaggi di programmazione più recenti (infatti, prima di essere elaborato, viene analizzato normalmente dal precompilatore ‘**cpp**’).

Oltre al normale commento indicato attraverso il punto esclamativo, si può utilizzare la forma del linguaggio C: ‘**/* ... */**’, segnalando così l’inizio e la fine del commento.

Direttiva	Descrizione
<code>#define <i>entità</i></code>	Definisce un’entità, ovvero una sorta di variabile booleana che conta solo in quanto esistente o meno.
<code>#ifdef <i>entità</i></code>	Se una certa entità è stata definita, esegue le righe seguenti fino a ‘ #endif ’.
<code>#ifndef <i>entità</i></code>	Se una certa entità non è stata definita, esegue le righe seguenti fino a ‘ #endif ’.
<code>#else</code>	Viene usato sia da ‘ #ifdef ’ che da ‘ #ifndef ’ per indicare le righe da eseguire in caso la condizione non si sia verificata.
<code>#endif</code>	Viene usato sia da ‘ #ifdef ’ che da ‘ #ifndef ’ per terminare una struttura condizionale.
<code>#include <i>file</i></code>	Include il contenuto del file indicato, in corrispondenza di quel punto.
<code><i>/* istruzioni */</i></code>	Commenta le istruzioni racchiuse in questo modo.

Quello che segue è l’esempio di un pezzo del contenuto di un file ‘`~/Xdefaults`’:

```
! Commentare la riga seguente se lo schermo è di grandi
! dimensioni.
#define SCHERMO_PICCOLO

#ifdef SCHERMO_PICCOLO
XTerm*geometry: =80x25+1+1
#else
XTerm*geometry: =100x40+1+1
#endif

! Mi piace la calcolatrice verde.
XCalc*background: green
! Voglio una barra del titolo differente.
XCalc*title: Calcolatrice
```

Come si vede, se viene dichiarata l'entità '**SCHERMO_PICCOLO**', viene definita una geometria normale per il programma '**XTerm**', altrimenti si usa una dimensione di 100×40. Per commentare la definizione dell'entità, si può fare come nell'esempio seguente:

```
! Commentare la riga seguente se lo schermo è di grandi
! dimensioni.
/* #define SCHERMO_PICCOLO */
```

29.1.2.3 Altre risorse

Il sistema di script che si occupa di attivare una sessione grafica, dovrebbe occuparsi di leggere altri file e aggiungere il loro contenuto alla definizione delle risorse, attraverso l'uso del programma '**xrdb**'. Dovrebbe trattarsi dei file contenuti nella directory '/etc/X11/Xresources/' ed eventualmente del file personale '~/.Xresources'.

Se esistono questi file, le risorse descritte al loro interno preval-



gono su quanto descritto invece all'interno di `‘/etc/X11/app-defaults/’` e `‘~/.Xdefaults’`.

29.1.2.4 Utilizzo di «xrdb»

«

`‘xrdb’`¹ (*X resources database*) permette di leggere o modificare le impostazioni delle risorse. Viene usato normalmente per leggere il contenuto di un file e aggiornare di conseguenza l'impostazione corrente delle risorse.

```
xrdb [opzioni] [file]
```

Si veda la pagina di manuale *xrdb(1)*.

Tabella 29.7. Alcune opzioni.

Opzione	Descrizione
<code>-merge</code>	Aggiunge le impostazioni ottenute dal file indicato.
<code>-query</code>	Permette di ottenere un listato delle impostazioni attive.

L'esempio seguente aggiunge il contenuto del file `‘./risorse’` alle impostazioni attuali delle risorse.

```
$ xrdb -merge ./risorse [Invio]
```

29.1.2.5 Opzione «-xrm»

«

```
-xrm risorsa
```

I programmi tradizionali X accettano anche questa opzione, eventualmente ripetuta più volte nella stessa riga di comando, per defini-

re una proprietà attraverso una risorsa. Ciò permette di definire delle caratteristiche senza intervenire su file di configurazione, senza dover richiamare il programma `'xrdp'` e senza interferire sugli altri programmi eventualmente avviati successivamente.

Segue la descrizione di alcuni esempi.

- `$ xcalc -xrm '*background: gold' [Invio]`

Avvia `'xcalc'` con un colore dorato per lo sfondo. Vengono usati gli apici singoli per evitare che la shell tenti di interpretare l'asterisco.

- `$ xcalc -xrm '*background: gold' ↵`
`↵ -xrm '*foreground: red' [Invio]`

Avvia `'xcalc'` con un colore dorato per lo sfondo e con un colore rosso per il primo piano.

29.2 Programmi di servizio tradizionali

Una serie di programmi di servizio tradizionali facilita e rende confortevole l'utilizzo di X. Le scrivanie grafiche come Gnome e KDE offrono molti programmi di servizio armonizzati e più gradevoli da usare; tuttavia, è importante conoscere anche l'uso dei programmi tradizionali.

29.2.1 Terminale

Il primo programma da dover conoscere quando si utilizza X è quello che consente di gestire un terminale a caratteri attraverso una finestra. Il programma tradizionale utilizzato per questo scopo è `'xterm'`.² In alternativa si possono usare programmi simili, che possono avere più o meno funzionalità di quello originale.

Il comportamento di un terminale a finestra non è esattamente uguale a quello di una console; ci si accorge subito che i soliti programmi non rispondono alla tastiera nello stesso modo cui si è abituati. Quando ciò accade, vale almeno la pena di provare tutti i programmi di terminale a finestra a disposizione, per determinare quale si comporta nel modo più confacente alle proprie esigenze. La sintassi semplificata per l'uso di **'xterm'** è la seguente:

```
xterm [opzioni] [-e programma [opzioni]]
```

Quando il programma viene avviato senza l'opzione **'-e'**, viene eseguito quanto contenuto nella variabile **SHELL** e se manca viene utilizzato **'/bin/sh'**.

Se invece si utilizza l'opzione **'-e'**, si può specificare il programma da eseguire nella finestra. Ciò può essere utile per preparare dei comandi già pronti all'interno di menù di altri programmi o del gestore di finestre stesso.

Una cosa importante da sottolineare è che le dimensioni della geometria di una finestra di terminale si esprimono in caratteri e non in punti come si fa di solito.

Infine, con l'opzione **'-ls'**, si ottiene una shell di *login*, mentre con **'+ls'** no.

Segue la descrizione di alcuni esempi.

- `$ xterm -font 5x8 -geometry =132x30+0+0 [Invio]`

Avvia una finestra di terminale utilizzando caratteri molto piccoli con una dimensione di 30 righe per 132 colonne, posizionata a partire dall'angolo superiore sinistro dello schermo.

- `$ xterm -e top` [Invio]

Avvia una finestra di terminale di dimensioni predefinite (80×25) con il programma `'top'`. Quando l'esecuzione di `'top'` viene conclusa, la finestra del terminale si chiude.

- `$ xterm -ls` [Invio]

Avvia una finestra di terminale di dimensioni predefinite, con una shell di *login*.

- `$ xterm -en UTF-8` [Invio]

Avvia una finestra di terminale in cui si utilizza la codifica dei caratteri UTF-8.

Dopo `'xterm'`, il programma più importante è `'mlterm'`,³ che si distingue per la sua migliore predisposizione alla rappresentazione di testi in lingue differenti, modificando automaticamente anche la direzione del testo, in base al linguaggio usato:

```
mlterm [opzioni] [-e programma [opzioni]]
```

Il funzionamento e l'utilizzo è equivalente a quello di `'xterm'`, con la differenza che le opzioni non sono sempre compatibili. In particolare, per ottenere una shell di *login*, occorre usare l'opzione `'--ls=true'`, mentre per assicurarsi che non lo sia, si usa l'opzione `'--ls=false'`.

29.2.2 Clipboard

Il server X offre un supporto modesto alla gestione delle operazioni con cui si taglia, copia e incolla: si tratta esclusivamente del-

le stringhe (alfanumeriche), per cui tutto si limita alla possibilità di copiare una parte di testo da una finestra di terminale a un'altra.

L'operazione di copia avviene utilizzando il mouse, premendo il tasto sinistro e trascinando in modo da evidenziare il testo desiderato. Per incollare in un'altra applicazione occorre fare in modo che questa passi in primo piano (cioè che la sua finestra diventi quella attiva), poi basta premere il secondo tasto (normalmente è quello centrale) e il testo viene inserito come se venisse digitato, a partire dalla posizione del cursore.

Quando non si dispone di un mouse a tre tasti, oppure se il tasto centrale non funziona, si ottiene la funzione del tasto centrale con la pressione simultanea dei due tasti funzionanti.

Per incollare del testo all'interno di un'applicazione VI, occorre prima attivare la modalità di inserimento, altrimenti VI utilizza il testo incollato come una serie di comandi. Lo stesso ragionamento vale ovviamente anche per altri programmi che possono utilizzare i caratteri normali sia come testo da inserire che come comandi da eseguire.

Ci sono anche altri modi per evidenziare un testo, ma quello che conta è che il testo selezionato per la copia deve rimanere evidenziato fino al momento in cui si intende incollare quel testo.

Molti programmi sono in grado di utilizzare questo servizio offerto da X, ma non tutti. Alcuni programmi hanno la necessità di gestire in proprio le funzionalità relative alle operazioni per tagliare, copiare e incollare, soprattutto quando si tratta di testo formattato,

immagini e altro. In questi casi, vengono utilizzati dei meccanismi di comunicazione tra i processi indipendenti dal sistema grafico. Di conseguenza, possono comunicare tra loro solo i processi predisposti per quel particolare sistema di comunicazione. Questo dovrebbe chiarire il motivo per cui il trasferimento di informazioni tra un'applicazione e l'altra, attraverso operazioni per tagliare, copiare e incollare, funziona solo in alcune situazioni e tra particolari gruppi di programmi.

29.2.2.1 Utilizzo di «xclipboard»

Il programma '**xclipboard**'⁴ facilita l'utilizzo del servizio per tagliare, copiare e incollare, fornito dal server X. Si tratta di una serie di pagine su cui è possibile scrivere e incollare del testo attraverso il meccanismo normale di X.

```
xclipboard [opzioni]
```

Figura 29.8. Il programma `'xclipboard'` permette di utilizzare un'area transitoria per tagliare, copiare e incollare.



Sotto questo aspetto si tratta di niente di più che una specie di programma per la creazione e modifica di testo. Tuttavia, l'accorgimento della gestione di pagine separate lo rende più pratico per questo scopo.

Il programma mostra un menù molto semplice composto da alcune voci all'interno di pulsanti grafici:

QUIT	termina l'esecuzione;
DELETE	svuota il contenuto della pagina attiva;
NEW	crea una nuova pagina;
SAVE	salva la pagina corrente in un file di testo normale;
NEXT	visualizza la pagina successiva;
PREV	visualizza la pagina precedente.

Per incollare del testo in un'applicazione, utilizzando quanto conservato con questo programma, si deve selezionare la pagina che inte-

ressa e poi si deve evidenziare il testo desiderato. Quindi si incolla nel modo solito.

29.2.3 Caratteri

Nel capitolo introduttivo a X si accenna all'organizzazione dei nomi dei caratteri da stampa usati per la visualizzazione sullo schermo. In particolare, quando i programmi fanno riferimento a un carattere (un tipo di carattere), è consentito normalmente l'uso di simboli jolly (o metacaratteri): l'asterisco e il punto interrogativo. Questi simboli hanno lo stesso significato che gli si attribuisce quando vengono usati per i nomi dei file: l'asterisco corrisponde a qualunque sequenza di caratteri, mentre il punto interrogativo corrisponde a un solo carattere qualsiasi.

Quando attraverso la riga di comando si deve fare riferimento a un modello, cioè un nome che fa uso di simboli jolly, è bene ricordare che la shell interpreta questi simboli se non vengono protetti. Nel caso delle shell derivate da quella di Bourne, basta racchiudere il nome tra apici singoli.

La maggior parte dei programmi, quando deve fare riferimento a un carattere tipografico attraverso la riga di comando, riconosce l'opzione '**-font *tipo_di_carattere***' (abbreviabile anche con '**-fn**').

29.2.3.1 Utilizzo di «xlsfonts»

Il programma '**xlsfonts**'⁵ elenca i caratteri tipografici a disposizione in base a quanto specificato attraverso le opzioni. L'opzione più importante è '**-font *modello***', con la quale è possibile indicare un gruppo di caratteri.

```
xlsfonts [opzioni]
```

Segue la descrizione di alcuni esempi.

- \$ **xlsfonts** [Invio]

Elenca tutti i caratteri tipografici a disposizione.

- \$ **xlsfonts -font '*'** [Invio]

Esattamente come nell'esempio precedente, solo che viene indicato espressamente un modello che si riferisce a tutti i tipi di carattere.

- \$ **xlsfonts -font '-courier-*** [Invio]

Elenca tutti i tipi di carattere il cui nome inizia per **'-courier-'**.

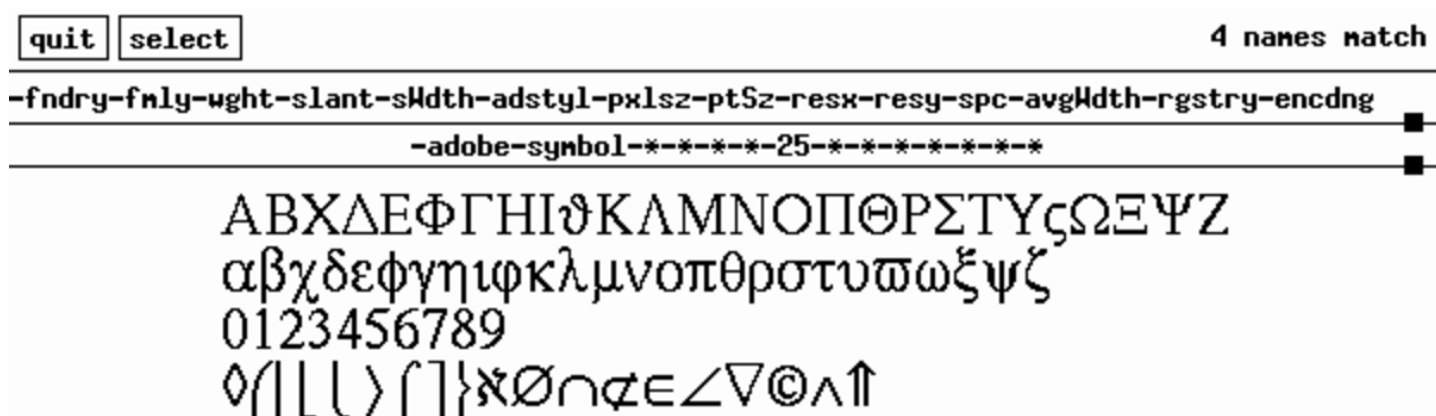
29.2.3.2 Utilizzo di «xfontsel»

«

Il programma **'xfontsel'**⁶ consente di conoscere quali sono i tipi di carattere a disposizione. È comodo da usare, soprattutto perché fornisce la possibilità di selezionare il tipo di carattere attraverso la specificazione delle caratteristiche desiderate per mezzo di un sistema di menù.

```
xfontsel [opzioni]
```

Figura 29.10. Attraverso ‘**xfontsel**’ si possono visualizzare i tipi di carattere a disposizione.

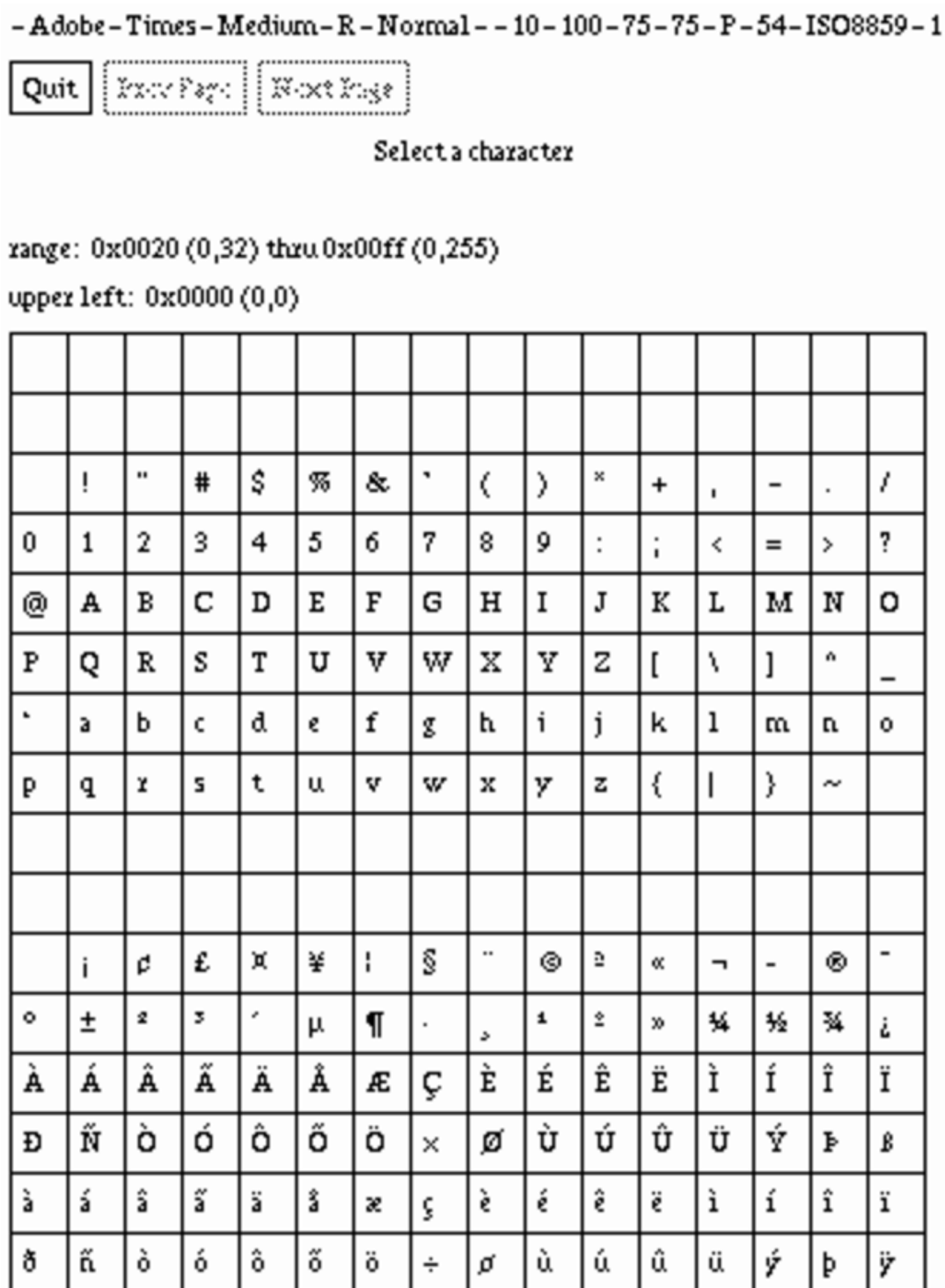


29.2.3.3 Utilizzo di «xfd»

Il programma ‘**xfd**’⁷ visualizza l’aspetto e la codifica di carattere tipografico che deve essere determinato obbligatoriamente dalle opzioni. Di conseguenza, l’utilizzo dell’opzione ‘**-font**’ è obbligatoria.

```
xfd [opzioni] -font tipo_di_carattere
```

Figura 29.11. Attraverso 'x_fd' si può visualizzare l'insieme di caratteri corrispondente a un certo tipo di carattere.



29.2.4 Informazioni sulle finestre e sul server

Le informazioni sullo stato di una finestra possono essere utili sia a titolo diagnostico, sia per poter riprodurre le stesse condizioni attraverso la configurazione di opzioni o di risorse.

Le informazioni su un server possono essere interessanti, in particolare quando vengono richieste a distanza.

29.2.4.1 Utilizzo di «xwininfo»

Il programma `'xwininfo'`⁸ permette di avere tutte le notizie possibili su una finestra determinata. Emette il risultato attraverso lo standard output, quindi conviene avviare questo programma da una finestra di terminale, se non si intende ridirigere l'output.

```
xwininfo [opzioni]
```

Tabella 29.12. Alcune opzioni.

Opzione	Descrizione
<code>-id</code> <i>identificatore_della_finestra</i>	Permette di specificare la finestra della quale si vogliono le informazioni, attraverso il codice esadecimale che la identifica.
<code>-root</code>	Emette le informazioni sulla finestra principale cioè la superficie grafica su cui si collocano le finestre normali.
<code>-int</code>	Richiede che gli identificatori delle finestre siano emessi in forma decimale, mentre normalmente vengono mostrati in esadecimale.

Opzione	Descrizione
-children	Emette l'indicazione della finestra principale, di quella genitrice e delle figlie di quella specificata. Permette quindi di avere una visione della dipendenza che c'è tra le finestre, con particolare attenzione alle figlie.
-tree	Emette l'indicazione della finestra principale, di quella genitrice, delle figlie e delle successive, ricorsivamente. Funziona in maniera simile all'opzione '-children', con la differenza che mostra tutte le discendenze.
-stats	Emette molte informazioni riferite alla finestra. Corrisponde al comportamento predefinito, quando non si specificano opzioni.
-all	Emette tutte le informazioni possibili.

Ecco come si comporta '**xwininfo**':

```
$ xwininfo [Invio]
```

```
xwininfo: Please select the window about which you  
would like information by clicking the  
mouse in that window.
```

Il programma invita a utilizzare il mouse per indicare una finestra della quale si vogliono conoscere le informazioni.

```
xwininfo: Window id: 0x2000002 "rxvt"  
  
Absolute upper-left X: 272  
Absolute upper-left Y: 165  
Relative upper-left X: 0  
Relative upper-left Y: 0  
Width: 494  
Height: 329  
Depth: 8  
Visual Class: PseudoColor  
Border width: 0  
Class: InputOutput  
Colormap: 0x26 (installed)  
Bit Gravity State: ForgetGravity  
Window Gravity State: NorthWestGravity  
Backing Store State: NotUseful  
Save Under State: no  
Map State: IsViewable  
Override Redirect State: no  
Corners: +272+165 -34+165 -34-106 +272-106  
-geometry 80x25-29+143
```

29.2.4.2 Utilizzo di «xdpyinfo»

Il programma ‘**xdpyinfo**’⁹ permette di avere tutte le informazioni possibili su un servente X particolare, eventualmente anche remoto. «

```
xdpyinfo [opzioni]
```

Opzione	Descrizione
-display <i>identificatore_del_servente</i>	Permette di definire esplicitamente le coordinate necessarie a raggiungere il servente che si desidera interrogare.

Si osservi l'esempio seguente:

```
$ xdpyinfo dinkel.brot.dg:0 [Invio]
```

```
name of display:      dinkel.brot.dg:0.0
version number:      11.0
vendor string:       The XFree86 Project, Inc
vendor release number: 3200
maximum request size: 4194300 bytes
motion buffer size:  256
bitmap unit, bit order, padding: 32, LSBFirst, 32
image byte order:    LSBFirst
number of supported pixmap formats: 2
supported pixmap formats:
    depth 1, bits_per_pixel 1, scanline_pad 32
    depth 8, bits_per_pixel 8, scanline_pad 32
keycode range:      minimum 9, maximum 117
focus: window 0x1800002, revert to Parent
number of extensions: 15
    BIG-REQUESTS
    DOUBLE-BUFFER
    MIT-SCREEN-SAVER
    MIT-SHM
    MIT-SUNDRY-NONSTANDARD
    RECORD
    SHAPE
    SYNC
    XC-MISC
    XFree86-DGA
    XFree86-Misc
    XFree86-VidModeExtension
    XInputExtension
```

XKEYBOARD

XTEST

default screen number: 0

number of screens: 1

screen #0:

dimensions: 800x600 pixels (271x203 millimeters)

resolution: 75x75 dots per inch

depths (2): 1, 8

root window id: 0x2a

depth of root window: 8 planes

number of colormaps: minimum 1, maximum 1

default colormap: 0x26

default number of colormap cells: 256

preallocated pixels: black 0, white 1

options: backing-store YES, save-unders YES

largest cursor: 64x64

current input event mask: 0x58003d

KeyPressMask ButtonPressMask ButtonReleaseMask

EnterWindowMask LeaveWindowMask SubstructureNotifyMask

SubstructureRedirectMask PropertyChangeMask

number of visuals: 6

default visual id: 0x20

visual:

visual id: 0x20

class: PseudoColor

depth: 8 planes

available colormap entries: 256

red, green, blue masks: 0x0, 0x0, 0x0

significant bits in color specification: 6 bits

visual:

visual id: 0x21

class: DirectColor

depth: 8 planes

available colormap entries: 8 per subfield

red, green, blue masks: 0x7, 0x38, 0xc0

significant bits in color specification: 6 bits

...

Il listato che si ottiene è molto lungo, ma le informazioni più importanti possono essere ritrovate nella prima parte. In particolare si nota la dimensione (800×600), la risoluzione (75×75 dpi), la profondità di colori (8 bit) e di conseguenza il numero di colori a disposizione (256):

```
screen #0:
  dimensions:      800x600 pixels (271x203 millimeters)
  resolution:      75x75 dots per inch
  depths (2):      1, 8
  root window id:  0x2a
  depth of root window: 8 planes
  number of colormaps: minimum 1, maximum 1
  default colormap: 0x26
  default number of colormap cells: 256
```

...

29.2.5 Impostazione dello schermo

«

La configurazione del funzionamento dello schermo riguarda il tipo di interazione tra l'utente e i programmi (tastiera, mouse, salva-schermo), oltre al tipo di superficie grafica, ovvero la finestra principale.

29.2.5.1 Utilizzo di «xset»

«

Il programma `'xset'`¹⁰ permette di definire e leggere una grande quantità di impostazioni che riguardano la stazione grafica (il server X). Le opzioni particolari di questo programma non utilizzano il trattino tradizionale, o quantomeno non nel modo solito. Quando l'utente che ha impostato la configurazione termina la sua sessione di lavoro, tutto torna al suo valore precedente.

```
xset [opzioni]
```

Tabella 29.18. Alcune opzioni.

Opzione	Descrizione
<code>-display <i>identificatore_del_servente</i></code>	Permette di definire esplicitamente le coordinate necessarie a raggiungere il servente su cui si desidera intervenire.
<code>b { <i>volume tono durata</i> } ← ↔ { on off }</code>	Definisce il suono dell'avvisatore acustico. Utilizzando gli argomenti 'on' oppure 'off' si attiva o si disattiva l'avvisatore acustico.
<code>m [{ <i>accelerazione</i> [<i>soglia</i>] } ← ↔ default]</code>	Permette di definire il valore di accelerazione e di soglia dello spostamento del dispositivo di puntamento. Se non viene indicato alcun valore, oppure se viene utilizzato l'argomento 'default', si ripristinano le impostazioni predefinite. L'accelerazione può essere stabilita indicando un numero intero che rappresenta un valore da moltiplicare alla velocità, quando questa supera il valore di soglia specificato. In pratica, se lo spostamento è al di sotto della soglia, il movimento del puntatore è lento, se questa viene superata, lo spostamento risulta accelerato del fattore di accelerazione indicato.

Opzione	Descrizione
r [on off]	Abilita o disabilita la ripetizione del tasto premuto a lungo.
s [{ <i>durata_inattività</i> ← ↔ [<i>durata_esposizione</i>] } ← ↔ <i>parola_chiave</i>]	Questa opzione permette l'utilizzo di due argomenti numerici o di una parola chiave. Lo scopo è quello di configurare il comportamento del salva-schermo. La durata di inattività rappresenta il tempo, in secondi, che deve trascorrere prima che si attivi il salva-schermo; la durata di esposizione riguarda il caso in cui si utilizzi un'immagine al posto dello schermo nero; rappresenta il tempo in cui questa immagine può rimanere ferma. Segue l'elenco e la descrizione delle parole chiave.
s default	Pone tutti i valori a quanto stabilito in modo predefinito.
s {on off}	Attiva o disattiva la modalità.
s {blank noblank}	In alcuni sistemi, non è possibile o non si desidera oscurare completamente lo schermo. 'blank' utilizza l'oscuramento, mentre 'noblank' mostra un'immagine e in tal caso si fa uso della durata di esposizione per sapere quanto tempo questa immagine può stare ferma.

Opzione	Descrizione
s activate	Attiva immediatamente il salva-schermo.
s reset	Disattiva il salva-schermo se è attivo.
q	Permette di ottenere le informazioni relative a tutte le impostazioni a cui può accedere 'xset' .

Segue la descrizione di alcuni esempi.

- `$ xset b 100 1000 100 [Invio]`

Imposta un suono acuto e breve per l'avvisatore acustico.

- `$ xset m 10 5 [Invio]`

Imposta un mouse veloce.

- `$ xset m 4 2 [Invio]`

Imposta un mouse normale.

- `$ xset s 30 s blank [Invio]`

Fissa la durata di attesa per l'attivazione del salva-schermo a 30 secondi e stabilisce che deve trattarsi di uno schermo nero.

- `$ xset q [Invio]`

Keyboard Control:

```
auto repeat:  on      key click percent:  0
                                LED mask:  00000000
auto repeat delay: 500    repeat rate:  5
auto repeating keys: 00feffffdffffbbf
                                fa9ffffffffffdf3d00
                                000000000000000000
                                000000000000000000
bell percent:  100    bell pitch:  200
                                bell duration: 1000
```

Pointer Control:

```
acceleration: 4/1    threshold:  4
```

Screen Saver:

```
prefer blanking: yes    allow exposures:  yes
timeout:  30    cycle:  1
suspend time: 900    off time:  1800
```

Colors:

```
default colormap: 0x26    BlackPixel:  0
                                WhitePixel:  1
```

Font Path:

```
/usr/lib/X11/fonts/misc/,/usr/lib/X11/fonts/75dpi/
```

```
Bug Mode: compatibility mode is disabled
```

Visualizza la configurazione corrente.

29.2.5.2 Utilizzo di «xsetroot»

«

Il programma ‘**xsetroot**’¹¹ permette di gestire le caratteristiche della finestra principale, ovvero la superficie grafica su cui si appoggiano le finestre normali.

```
xsetroot [opzioni]
```

Tabella 29.20. Alcune opzioni.

Opzione	Descrizione
<code>-display <i>identificatore_del_servente</i></code>	Permette di definire esplicitamente le coordinate necessarie a raggiungere il servente su cui si desidera intervenire.
<code>-def</code>	Ripristina l'impostazione predefinita.
<code>-cursor <i>file_puntatore file_maschera</i></code>	Permette di definire un'immagine diversa per il puntatore che appare quando questo si trova sulla superficie della finestra principale. Per realizzare questi file si può usare il programma 'bitmap' . L'immagine usata come primo argomento dell'opzione si «appoggia» sull'immagine usata come secondo. Se non si vogliono ottenere trasparenze o bordi, basta usare lo stesso file per il primo e il secondo argomento.
<code>-bitmap <i>immagine_bitmap</i></code>	Permette di definire una piccola immagine da usare ripetitivamente come fondale. Il formato dell'immagine è quello generato dal programma 'bitmap' .
<code>-gray</code> <code>-grey</code>	Rende lo sfondo grigio.
<code>-solid <i>colore</i></code>	Definisce il colore dello sfondo.

Segue la descrizione di alcuni esempi.

- `$ xsetroot -solid gold` [Invio]

Colora lo sfondo con la tinta ‘gold’.

- `$ xsetroot -cursor freccia.bitmap contorno.bitmap` [Invio]

Cambia l’aspetto del puntatore grafico, utilizzando il file ‘freccia.bitmap’ per il disegno nero, presumibilmente di una freccia, e il file ‘contorno.bitmap’ come maschera.

29.2.5.3 Disegno del puntatore del mouse

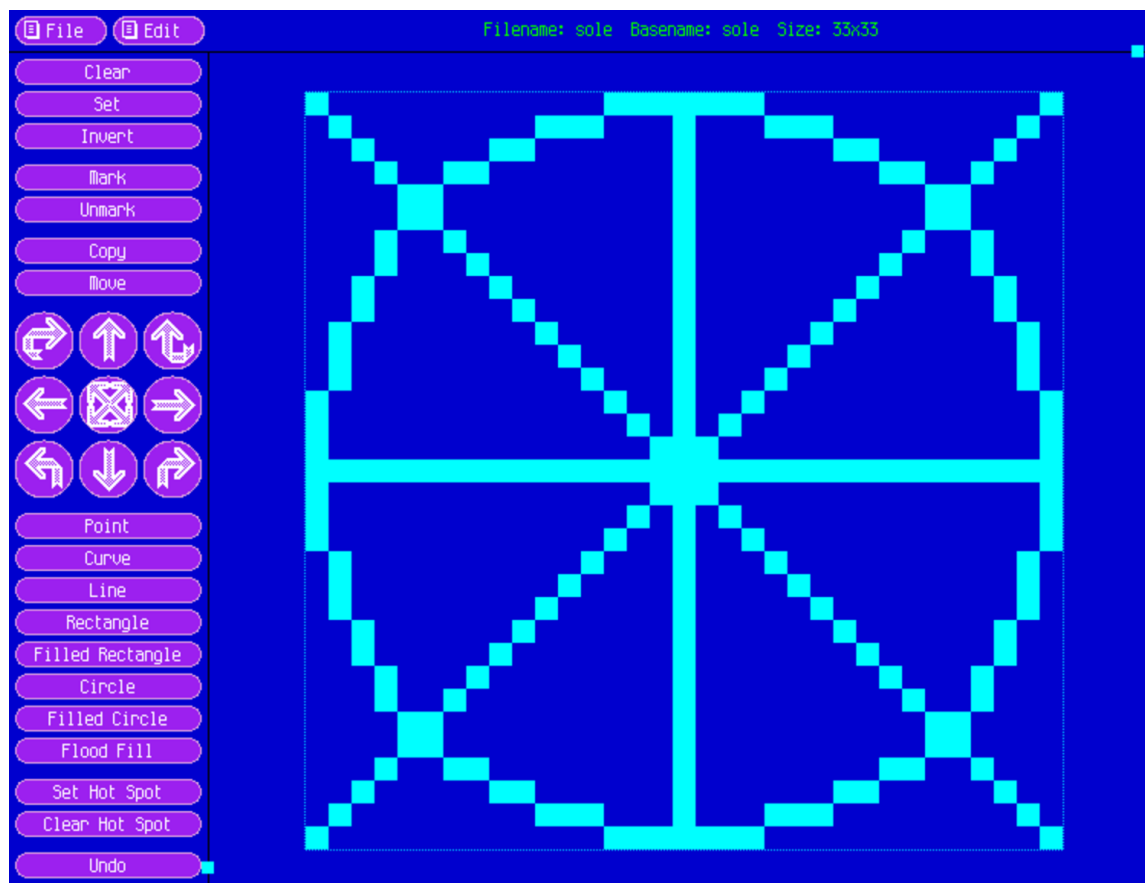
«

Attraverso il programma ‘**bitmap**’¹² è possibile disegnare l’immagine del puntatore grafico, secondo il formato previsto da ‘**xsetroot**’. Il programma si usa normalmente in modo interattivo, senza fornire argomenti:

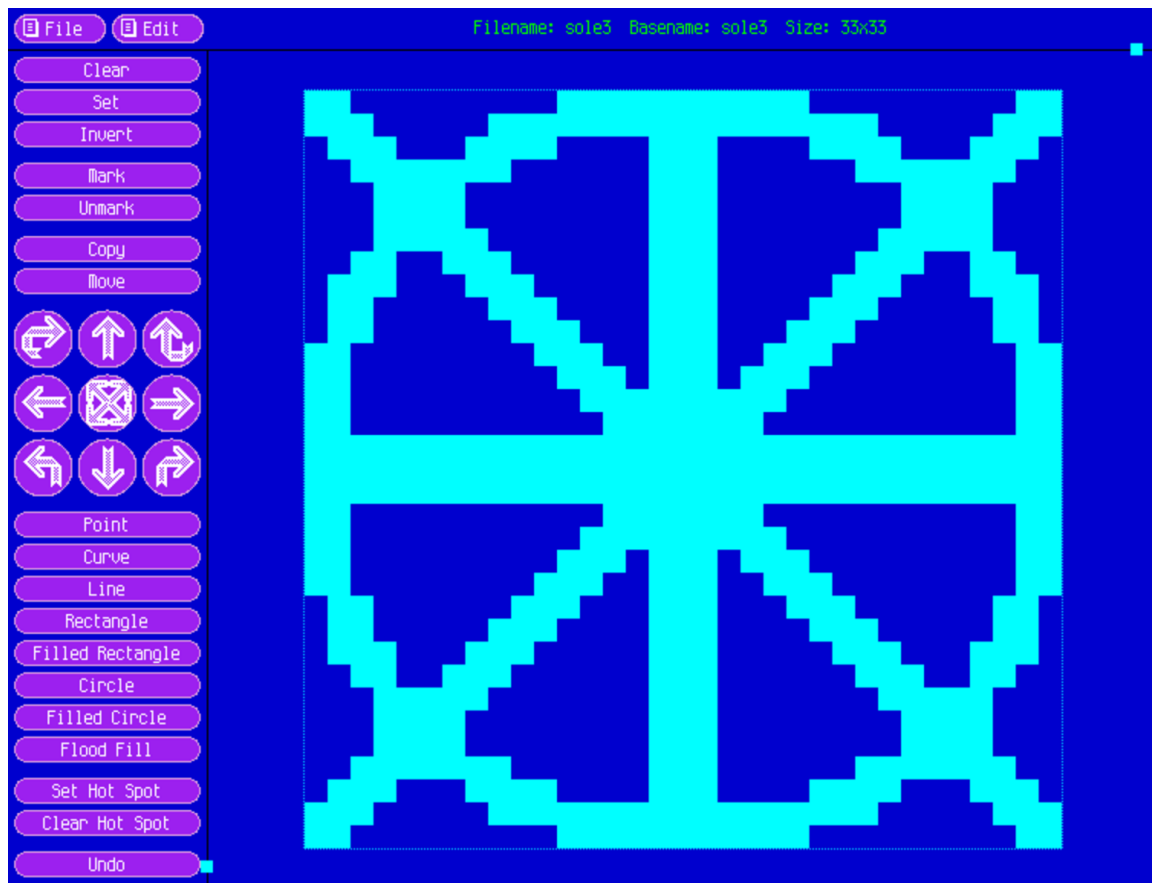
```
bitmap [opzioni] [file]
```

Il disegno che si può costruire è semplicemente un mosaico rettangolare di tessere nere o bianche; normalmente la superficie iniziale è di 16×16 punti, ma può essere modificata selezionando la voce *Resize* del menù *File*.

Figura 29.21. Il programma **'bitmap'** con un disegno di 33×33 punti.



L'immagine che si vede nella figura potrebbe essere usata per costruire un puntatore grafico. Per fare in modo che le linee appaiano nere, occorre utilizzare un'altra immagine, simile, da usare come maschera. Nella figura successiva si vede un'immagine che riproduce la stessa forma, con dei segni più spessi:



La seconda immagine viene usata come sfondo, di colore bianco, mentre la prima si posa sopra, con un tratto di colore nero:

```
$ xsetroot -cursor primo secondo [Invio]
```

Come si intende, il comando mostrato serve a utilizzare effettivamente il disegno come puntatore grafico, dove i nomi 'primo' e 'secondo' rappresentano rispettivamente il primo e il secondo file, realizzati con 'bitmap'.

29.2.6 Programmi di servizio vari

«

Alcuni programmi, per quanto semplici, sono di grande utilità, pertanto è opportuno conoscerne almeno l'esistenza.

Il controllo dell'utilizzo delle risorse di sistema può essere fatto attraverso 'xidle', 'xload' e 'xmem', che, rispettivamente, servono

per visualizzare il grafico: dell'inattività del sistema, del carico di sistema; della memoria disponibile.

```
xidle [opzioni]
```

```
xload [opzioni]
```

```
xmem [opzioni]
```

A volte si ha la necessità di concludere l'esecuzione di un'applicazione in modo più o meno violento perché questa è sfuggita al controllo. Di solito, per ottenere questo risultato si utilizza l'invio di un segnale attraverso una shell. Quando si tratta di applicazioni per X si può utilizzare una tecnica in più: si comunica al server di terminare la connessione con l'applicazione che si desidera concludere. Ciò si può ottenere attraverso una funzione fornita dal gestore di finestre o da un programma apposito: **'xkill'**.¹³

```
xkill [opzioni]
```

Il programma **'xkill'** permette di eliminare un programma funzionante in una finestra del sistema grafico X. Quando viene utilizzato senza argomenti, **'xkill'** trasforma il puntatore in un'immagine speciale (solitamente si tratta di un teschio nero) e permette di indicare direttamente la finestra da eliminare. Basta un clic e si ottiene il risultato.

Quando si utilizza prevalentemente il sistema grafico, si può avere

la necessità di essere avvisati in presenza di posta elettronica nella propria casella. Alcuni gestori di finestre forniscono già questo tipo di informazione, ma in mancanza di altro può essere utile **'xbiff'**:¹⁴

```
xbiff [opzioni]
```

Il programma **'xbiff'** si limita ad avvisare quando il file utilizzato per ricevere la posta elettronica risulta contenere qualcosa. Il programma è altamente configurabile, sia attraverso le opzioni, sia attraverso le risorse. In particolare, vale la pena di considerare l'opzione **'-file file'**, con cui si può indicare a **'xbiff'** di controllare un file differente rispetto a quello predefinito.

Figura 29.23. Il programma **'xbiff'** quando il file della posta elettronica è vuoto e quando ci sono dei messaggi.



Nelle sezioni successive si descrivono altri programmi di servizio un po' più importanti.

29.2.6.1 Utilizzo di «xclock»

«

Il programma **'xclock'**¹⁵ si occupa di visualizzare l'ora. Dal momento che è possibile visualizzare l'ora in modo digitale (numerico), l'opzione **'-font'** ha significato e può essere utile per cambiare l'aspetto dei caratteri.


```
xclock [opzioni]
```

Figura 29.24. `'xclock -digital -font '-adobe-times-bold-i-`

Sun Oct 12 16:06:46 1997

Alcuni gestori di finestre forniscono già un orologio attraverso i loro componenti; in questi casi non serve utilizzare `'xclock'`.¹⁶

Opzione	Descrizione
<code>-analog</code>	Mostra l'ora in modo analogico.
<code>-digital</code>	Mostra l'ora in forma numerica.

Risorsa	Descrizione
<code>*analog: {on off}</code>	Mostra l'ora in modo analogico (<code>'on'</code>) oppure digitale (<code>'off'</code>).

29.2.6.2 Utilizzo di «xcalc»

Il programma `'xcalc'`¹⁷ è una calcolatrice semplice e potente. Il suo funzionamento è abbastanza intuitivo, se non si desidera utilizzare la notazione polacca inversa, ma si tratta di un programma altamente configurabile ed eventualmente vale la pena di consultare la documentazione originale: *xcalc(1)*.

```
xcalc [opzioni]
```

Opzione	Descrizione
<code>-rpn</code>	Imposta l'aspetto e il funzionamento secondo la notazione polacca inversa.

Risorsa	Descrizione
<code>*rpn: {on off}</code>	Imposta o annulla l'aspetto e il funzionamento secondo la notazione polacca inversa.

29.3 Gestori di file

«

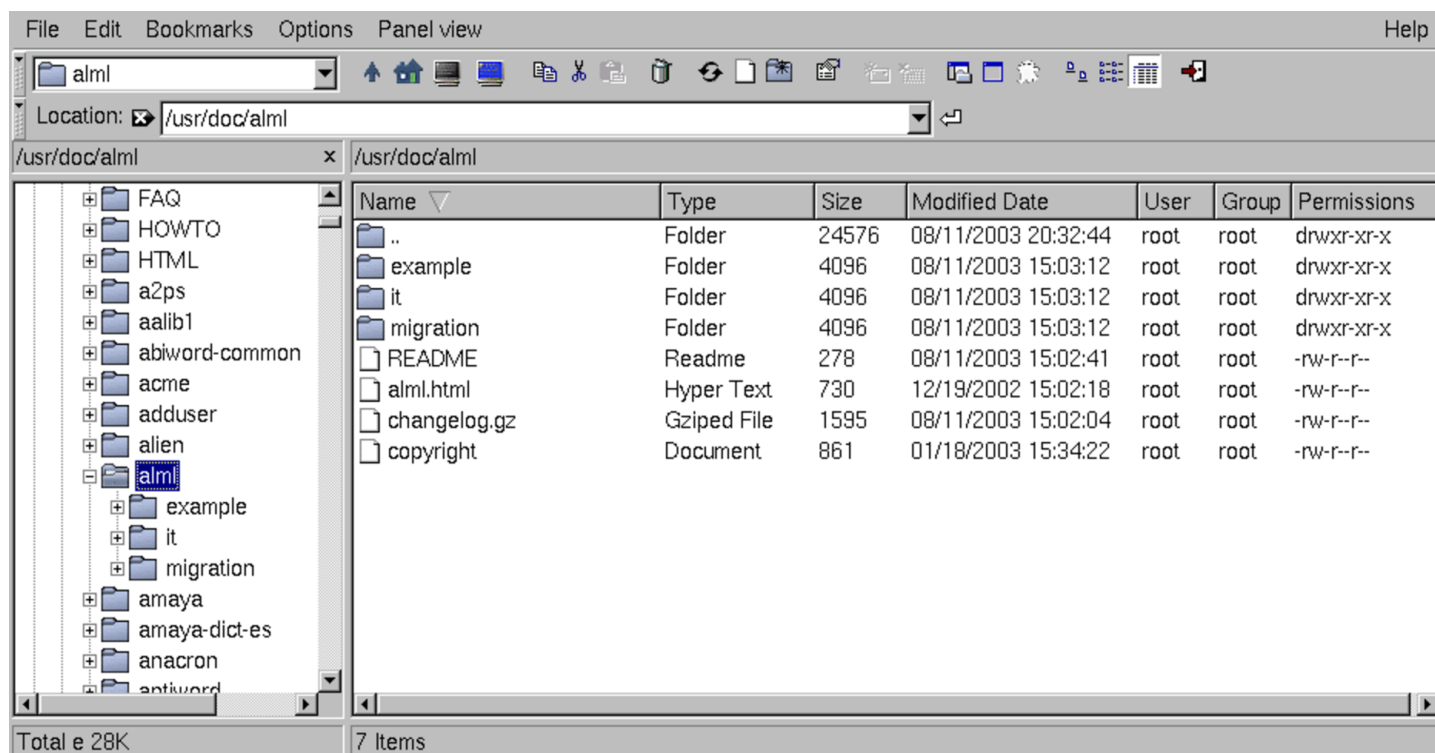
Un gestore di file (*file manager*) grafico può essere uno strumento molto utile se è configurato correttamente; diversamente, è meglio usare la finestra di terminale tradizionale. Il gestore di file è un programma che nella sua evoluzione, parte dalla gestione pura e semplice dei file contenuti nel file system locale, per arrivare a includere tutti i protocolli di comunicazione che in qualche modo consentono il trasferimento di file. Eventualmente, il punto massimo di evoluzione di un gestore di file consiste nell'integrazione delle funzionalità tipiche di un navigatore di Internet

29.3.1 XFE

«

XFE,¹⁸ ovvero *X file explorer*, è un gestore di file molto semplice e intuitivo. Quando si avvia l'eseguibile `'xfe'`, appare generalmente come nella figura 29.29.

Figura 29.29. XFE mentre è aperta la directory ‘/usr/doc/alml/’.



XFE si può configurare modificando il file ‘/etc/xfe/xferc’, cosa che diventa molto utile per stabilire esattamente quali programmi usare per aprire o modificare i file che hanno certe estensioni del nome. In alternativa, per questo gli utenti possono intervenire nel proprio file ‘~/ .xfe/xferc’.

Il file di configurazione è strutturato in modo abbastanza intuitivo. Si compone di sezioni, individuabili perché iniziano con un nome tra parentesi quadre, e di direttive. La modifica più probabile da apportare al file di configurazione riguarda la scelta dei programmi da avviare, a seconda delle azioni che si compiono con il mouse, quando il puntatore si trova sopra ai file con certe estensioni. Viene proposto un estratto di esempio di tale file:

```
[FILETYPES]
...
tex = "kedit,kedit,kedit;TeX Document;tex_32x32.png;tex_16x16.png;;"
dvi = "xdvi,xdvi,;DVI Document;dvi_32x32.png;dvi_16x16.png;;"
pdf = "xpdf,xpdf,;PDF Document;pdf_32x32.png;pdf_16x16.png;;"
ps = "gv,gv,;PostScript Document;ps_32x32.png;ps_16x16.png;;"
...
img = ",,;Image File;package_32x32.png;package_16x16.png;;"
iso = ",,;ISO9660 Image;package_32x32.png;package_16x16.png;;"
...
exe = "wine,,;Windows EXE;exe_32x32.png;exe_16x16.png;;"
...
```

Come si può intuire, alla sinistra di queste direttive appare l'estensione del file a cui si intende fare riferimento; quindi, dopo il segno '=', appare una stringa delimitata che contiene le indicazioni di come aprire il file e di come mostrarlo nell'elenco. In tale stringa possono apparire inizialmente tre nomi, separati da una virgola, i quali individuano i programmi da utilizzare, rispettivamente, per aprire (ovvero eseguire), visualizzare e modificare il file; successivamente viene collocata una descrizione del tipo di file e i file delle icone da mostrare.

Nell'esempio si può vedere che i file con estensione '.tex' vengono aperti, visualizzati e modificati sempre con lo stesso programma: '**kedit**'; nel caso invece dei file con estensione '.ps', è possibile solo l'apertura o la visualizzazione, utilizzando '**gv**' (Ghostview). Quando i file hanno estensione '.exe', è possibile solo aprirli (eseguirli), utilizzando il programma '**wine**'.

L'apertura di un file si ottiene con un clic doppio sul nome, mentre la selezione e la visualizzazione si ottengono facendo apparire un menù contestuale con il tasto destro del mouse.

29.3.2 Nautilus

Nautilus,¹⁹ come suggerisce il nome, intende divenire un «navigatore» generico, sia per il file system locale, sia per i protocolli di rete che consentono la gestione dei file in qualche modo.

```
nautilus [opzioni]
```

Nautilus è realizzato per integrarsi con l'ambiente di Gnome. All'avvio dell'eseguibile '**nautilus**' senza argomenti, si ottiene normalmente una finestra con l'elenco del contenuto della propria directory personale, assieme a una finestra che dovrebbe sostituirsi alla finestra principale, ovvero alla superficie grafica di fondo, nella quale appaiono delle icone associate a delle funzioni particolari. In generale, quando si usa Nautilus al di fuori del contesto di Gnome, conviene usare l'opzione '**--no-desktop**', per non avviare la finestra con le icone che molto probabilmente è del tutto inutile.

Figura 29.31. Nautilus mentre si accede alla directory personale dell'utente.

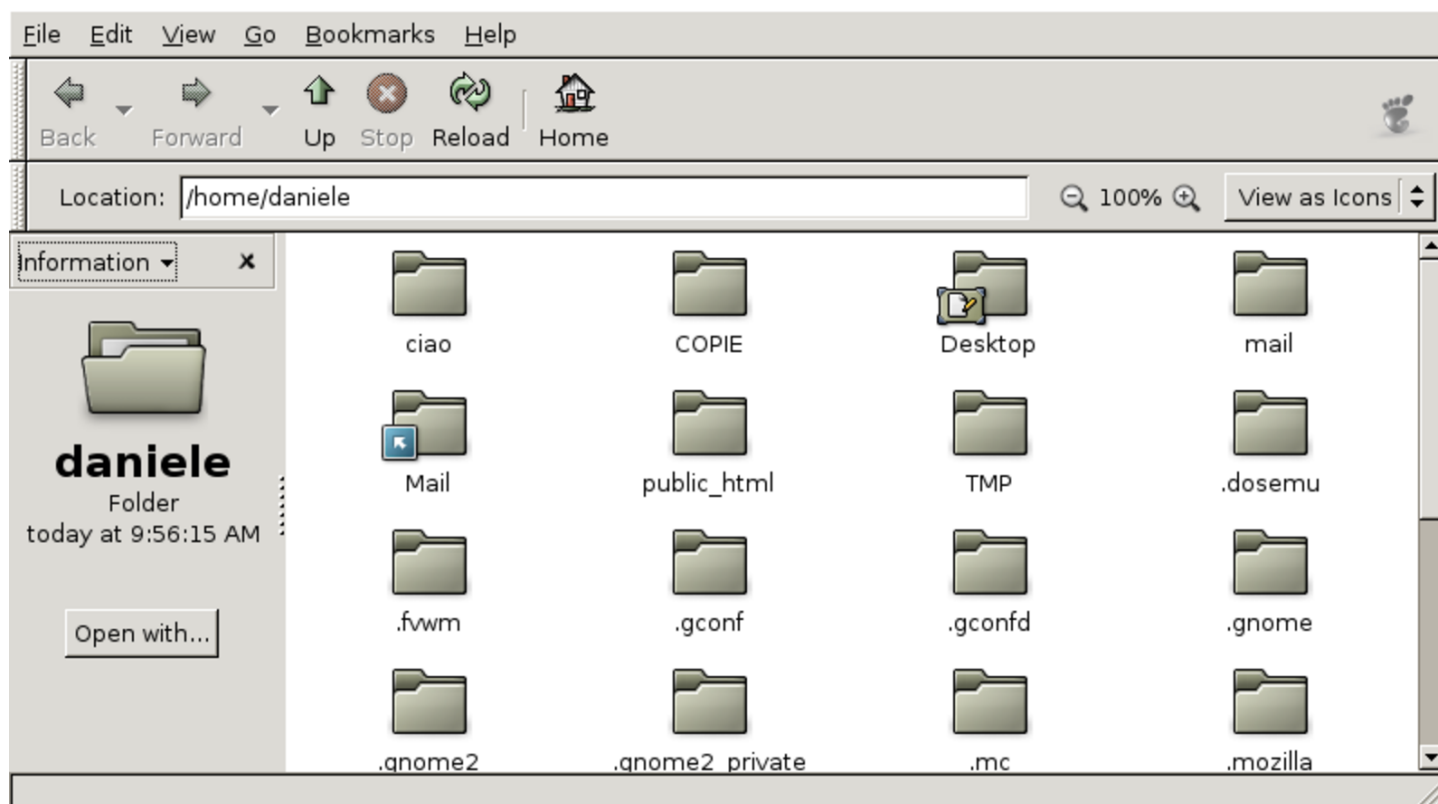
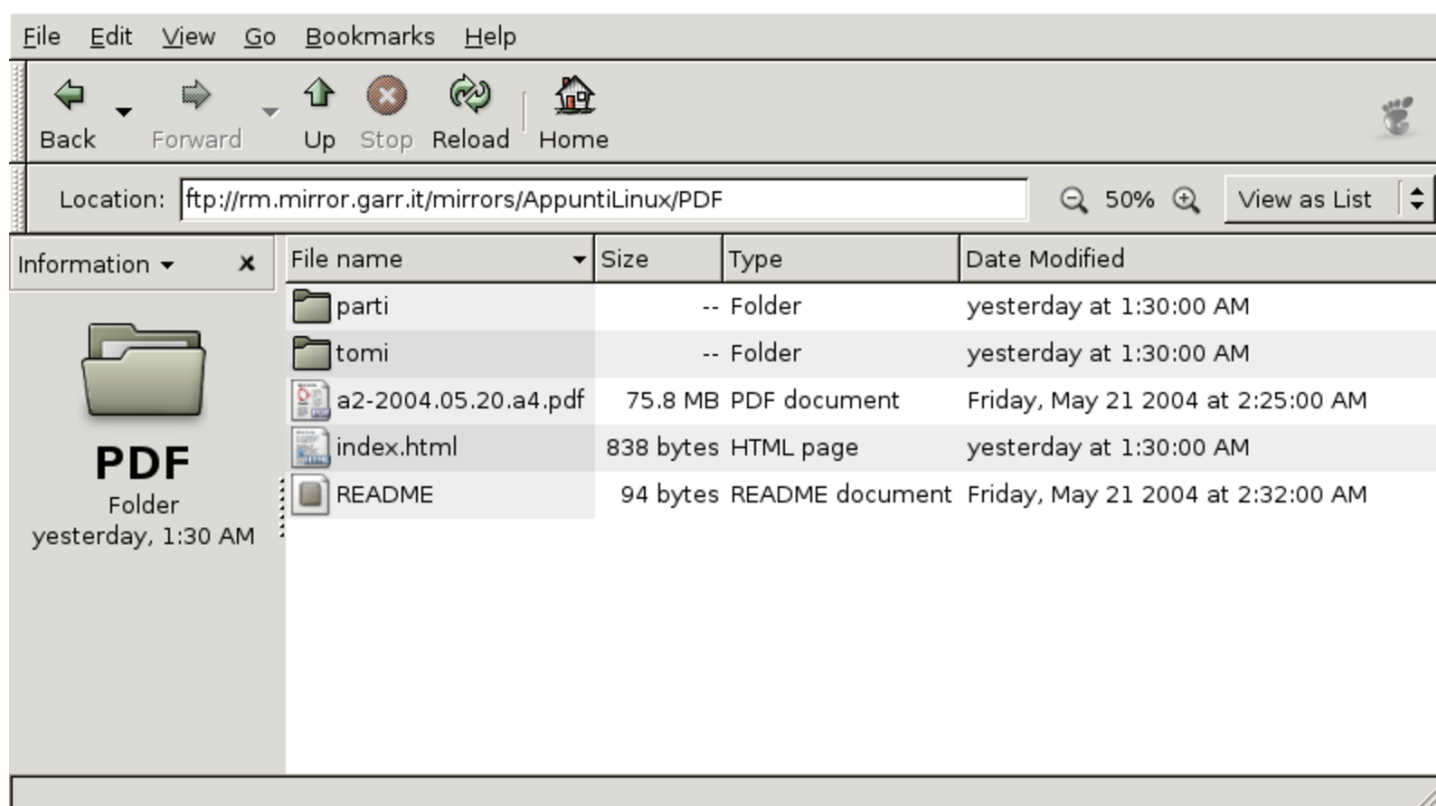


Figura 29.32. Nautilus mentre si accede a un servizio FTP anonimo.



29.3.3 Konqueror

Konqueror,²⁰ è un navigatore generico, completo, in grado di accedere al file system locale e ad altri protocolli di rete, compresa la capacità di visualizzare documenti HTML.

```
konqueror [opzioni] [uri]
```

Konqueror è inteso come il navigatore generico per l'ambiente KDE. All'avvio dell'eseguibile '**konqueror**' senza argomenti, si ottiene normalmente una finestra con l'elenco del contenuto della propria directory personale, altrimenti è possibile indicare un indirizzo URI o anche solo un percorso del file system locale.

Figura 29.33. Konqueror mentre si accede alla directory personale dell'utente.

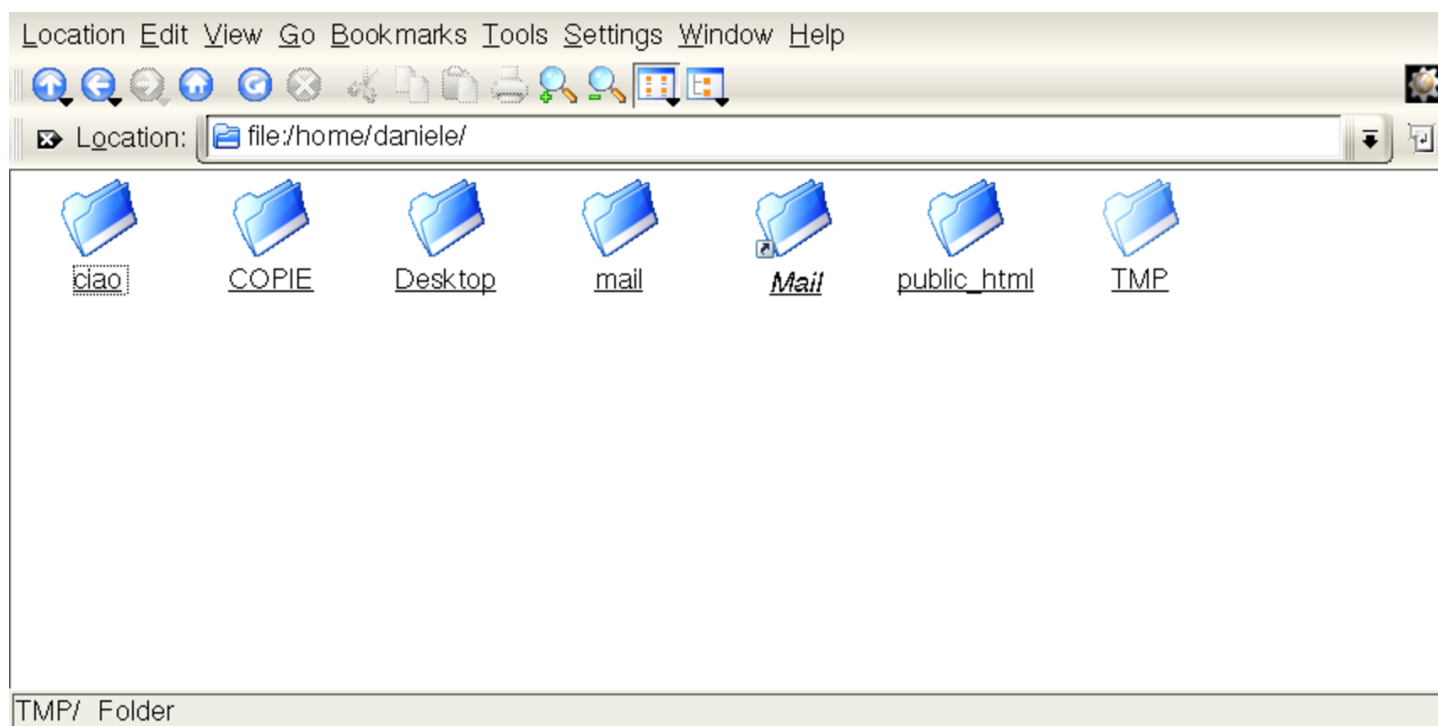
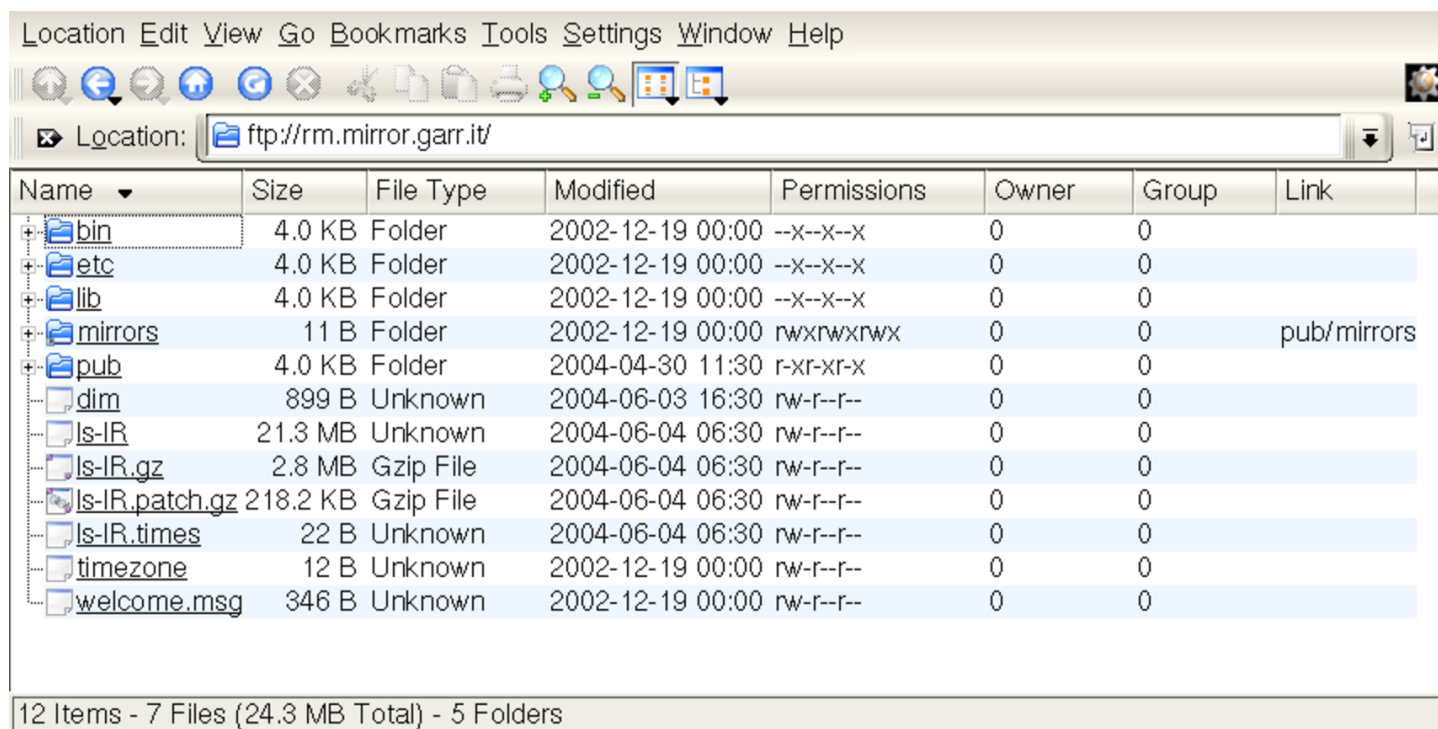


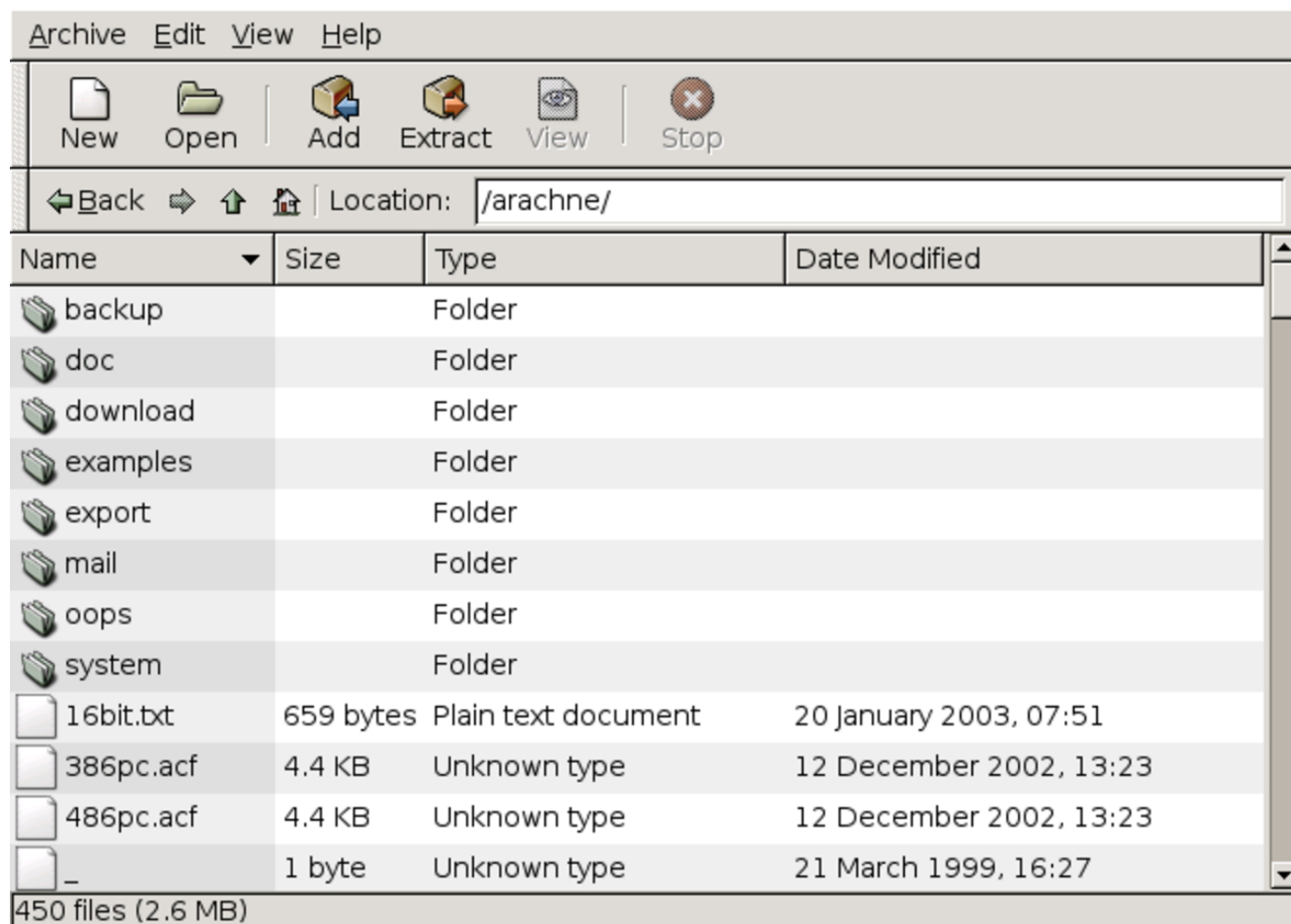
Figura 29.34. Konqueror mentre si accede a un servizio FTP anonimo.



29.3.4 File-roller

File-roller²¹ è un programma che ha lo scopo di facilitare l'accesso ad archivi compressi e la loro realizzazione, quando non si vogliono usare gli strumenti tradizionali (a riga di comando).

Figura 29.35. L'aspetto di File-roller dopo l'apertura di un file '.zip'.



L'utilità maggiore nell'uso di File-roller sta nell'abbinamento con programmi come Nautilus, che lo avviano automaticamente quando il contesto lo suggerisce.

29.4 Accesso remoto tradizionale

«

Per accesso remoto «tradizionale», si intende qui ciò che si può fare, per esempio con il protocollo TELNET e con SSH. I programmi per l'utilizzo di tali protocolli funzionano generalmente senza grafica, tuttavia tendono ad apparire anche programmi realizzati specificatamente per la grafica. Questo capitolo intende far conoscere l'esistenza di tali programmi, senza entrare nella descrizione dei protocolli utilizzati.

29.4.1 Putty

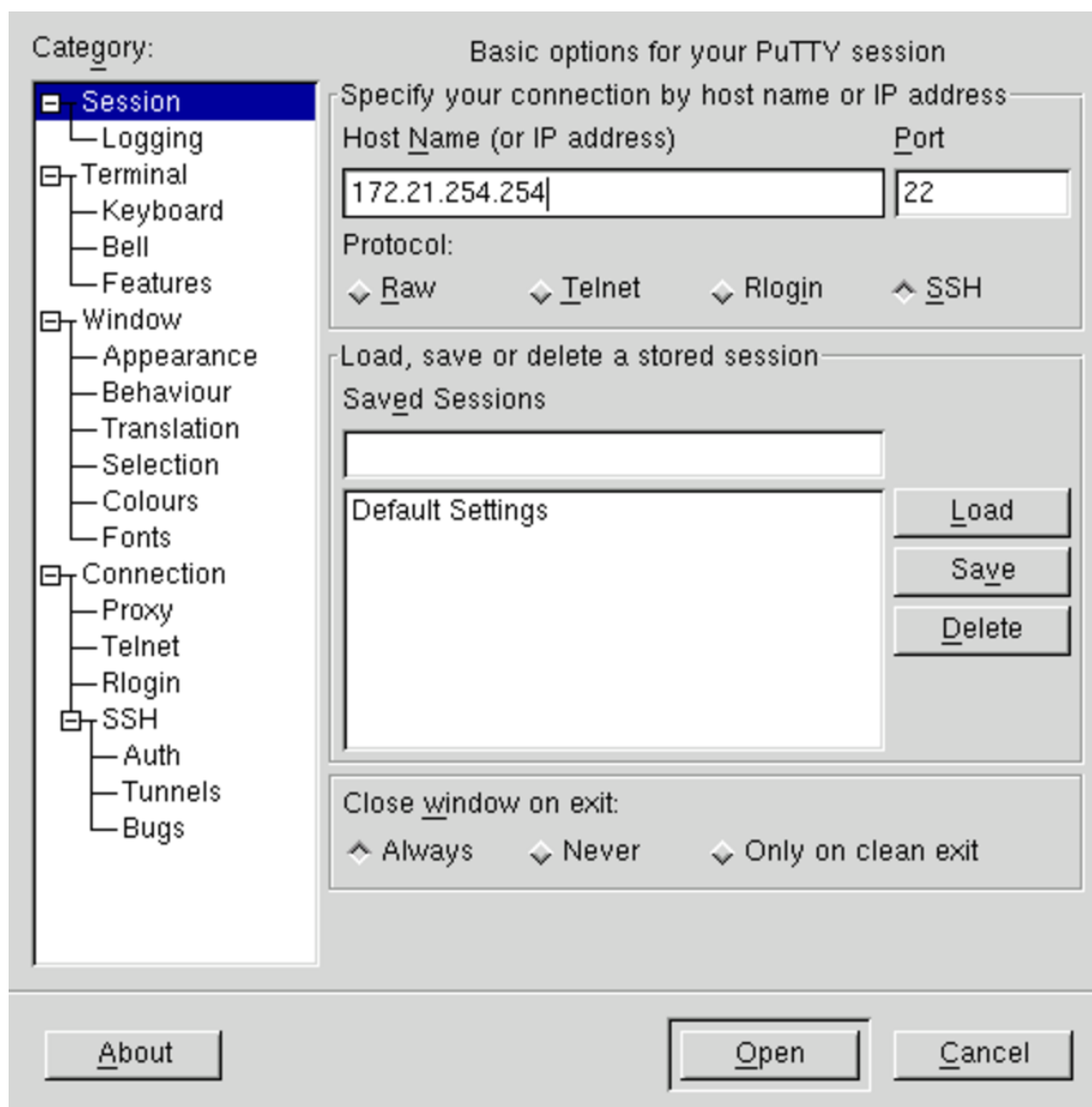
«

Putty²² è un programma abbastanza noto nell'ambiente MS-Windows, perché può essere utilizzato come programma cliente per accedere a un server TELNET (sezione 36.8) o SSH (si veda OpenSSH nella sezione 44.7).

```
putty [opzioni] [nodo]
```

Putty si avvia attraverso il programma eseguibile '**putty**', generalmente senza bisogno di specificare argomenti, perché anche l'indirizzo o il nome dell'elaboratore remoto da raggiungere può essere indicato attraverso una finestra di dialogo. Tuttavia, per usare Putty, si richiede almeno la conoscenza elementare nell'uso dei programmi tradizionali che svolgono lo stesso genere di lavoro.

Figura 29.36. Putty avviato senza argomenti particolari.



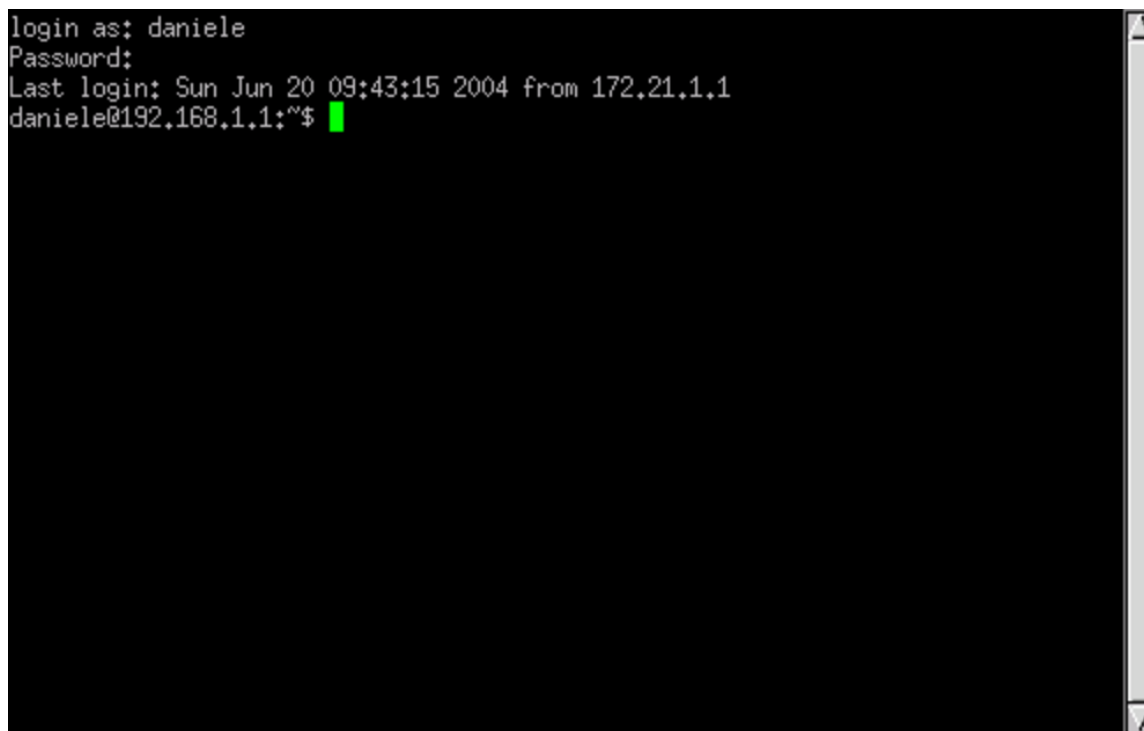
La figura mostra Putty appena dopo l'avvio, senza inserire opzioni o argomenti particolari. Si nota comunque che l'utente ha già inserito l'indirizzo IPv4 che intende raggiungere, dove si aspetta di intrattenere una connessione attraverso il protocollo SSH (porta 22).

Nella porzione sinistra della finestra di dialogo, appare un albero di voci che fanno riferimento alla configurazione di Putty; di solito non serve intervenire, per lasciare le impostazioni predefinite.

Quando si è pronti per il collegamento basta selezionare il pulsante

grafico [OPEN](#).

Figura 29.37. Putty al lavoro durante una connessione SSH.



```
login as: daniele
Password:
Last login: Sun Jun 20 09:43:15 2004 from 172.21.1.1
daniele@192.168.1.1:~$
```

29.5 Monitor di vario tipo

«

Il pacchetto tradizionale di programmi di servizio di X include qualche piccola applicazione per visualizzare lo stato di utilizzo del sistema e della memoria (`xmem`, `xload` e `xidle`). Eventualmente, altri programmi possono essere interessanti per avere una visione grafica di ciò che accade nel proprio elaboratore.

29.5.1 Gmemusage

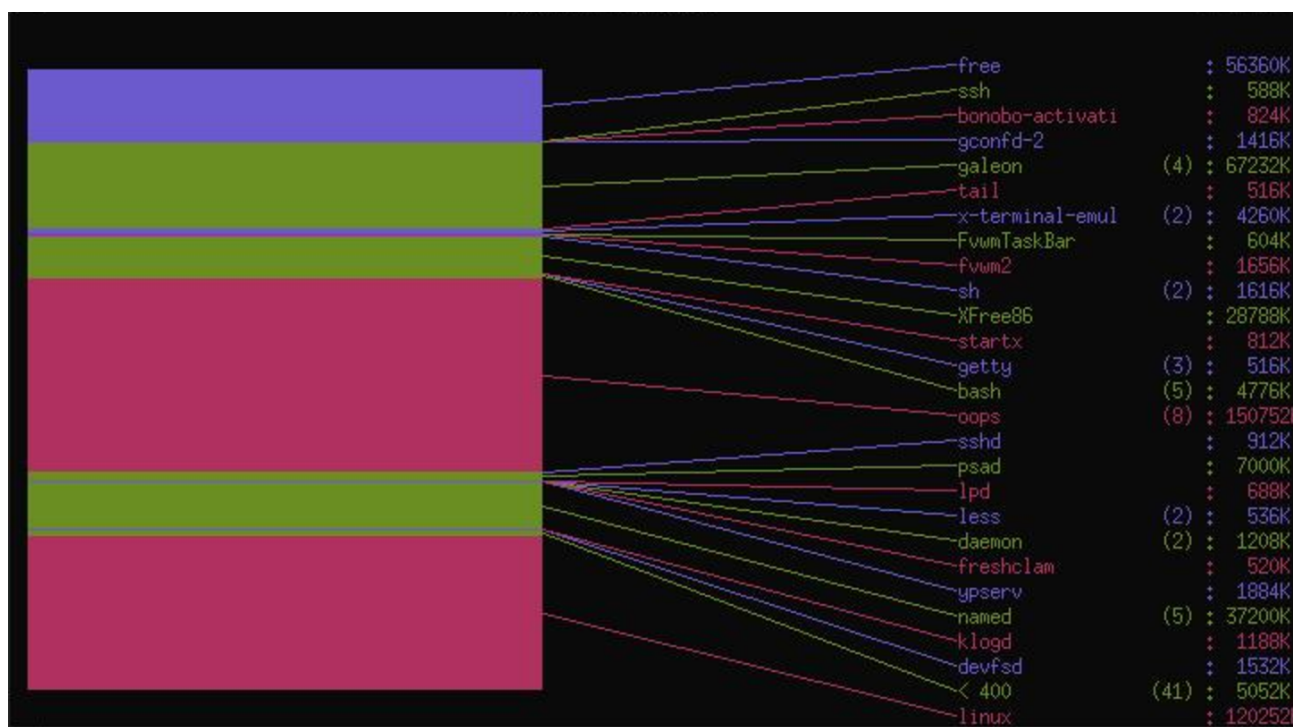
«

Gmemusage²³ è un programma che mostra graficamente l'utilizzo della memoria da parte dei processi elaborativi in funzione nel proprio sistema:

```
gmemusage [opzioni]
```

Attraverso le opzioni è possibile intervenire, tra le altre cose, sulla scelta dei colori, dei caratteri e delle dimensioni iniziali della finestra. Si può controllare anche l'entità minima di memoria da considerare, ma questo si può modificare anche durante il funzionamento del programma, con l'uso dei tasti [*freccia-su*] e [*freccia-giù*]. La figura successiva mostra come si presenta il programma avviando l'eseguibile **'gmemusage'** senza opzioni.

Figura 29.38. Gmemusage durante il funzionamento, in condizioni normali.



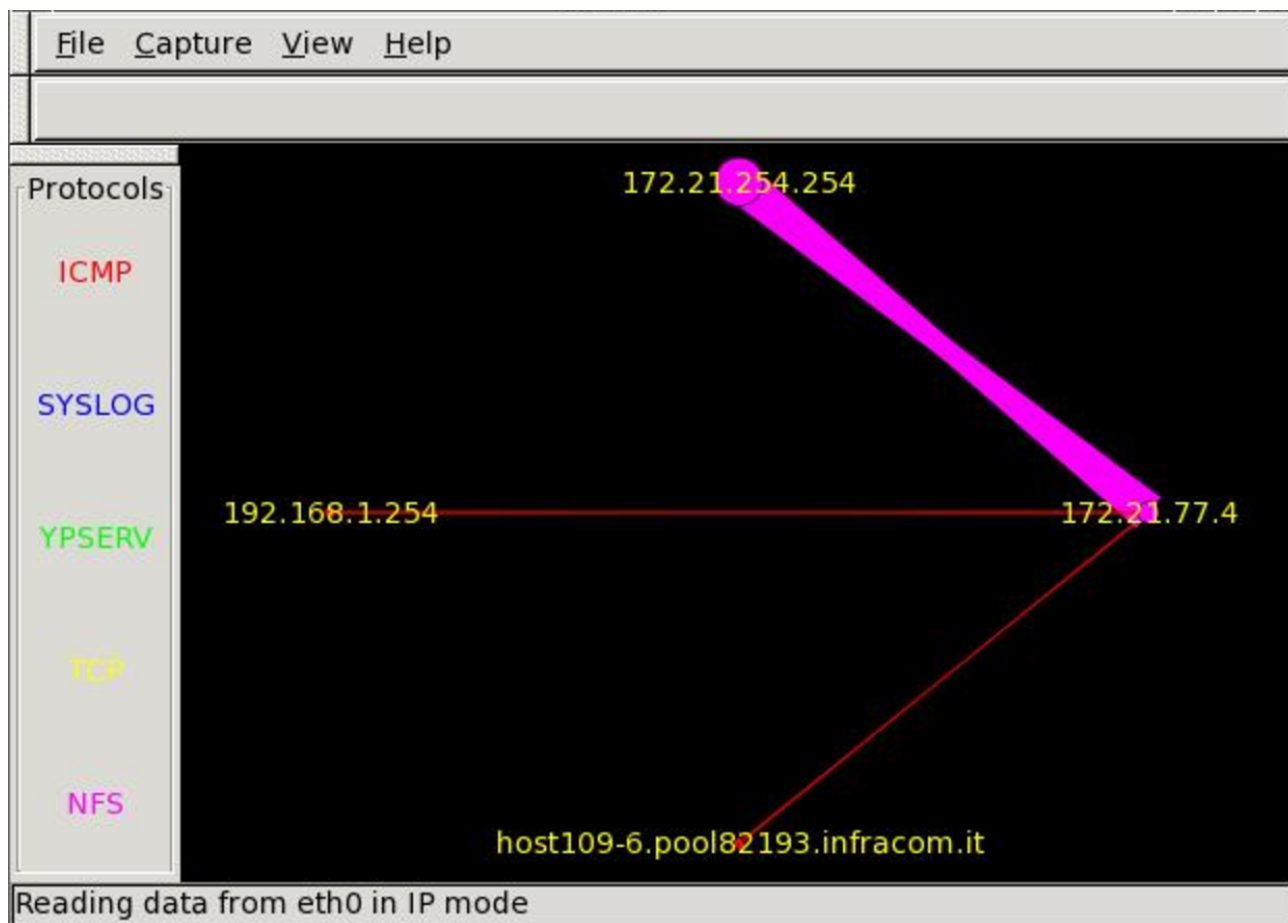
29.5.2 Etherape

Etherape²⁴ è un programma che mostra graficamente il traffico di rete che coinvolge, direttamente o indirettamente, il proprio elaboratore. Non si tratta di uno strumento sofisticato di controllo del traffico, ma di qualcosa che dà l'idea dell'entità di questo.

etherape [*opzioni*]

Etherape ha bisogno di accedere direttamente all'interfaccia di rete, pertanto si usa generalmente con i privilegi dell'utente **'root'**.

Figura 29.39. Etherape durante il funzionamento. Si vede in particolare un traffico ICMP, che in questo caso è di colore blu, e un traffico NFS, che appare di colore rosa.



29.6 Applicativi comuni per l'ufficio

«

Tra i programmi per l'automazione dell'ufficio, il più importante è LibreOffice (oppure Openoffice.org, se si preferisce); tuttavia, anche altri programmi per la scrittura, il foglio elettronico e il disegno, possono essere utili in certe circostanze. Quindi, programmi come

Gnumeric o AbiWord, possono essere utili in certe circostanze, anche se non se ne intende fare un uso regolare, pertanto è bene essere a conoscenza della loro disponibilità.

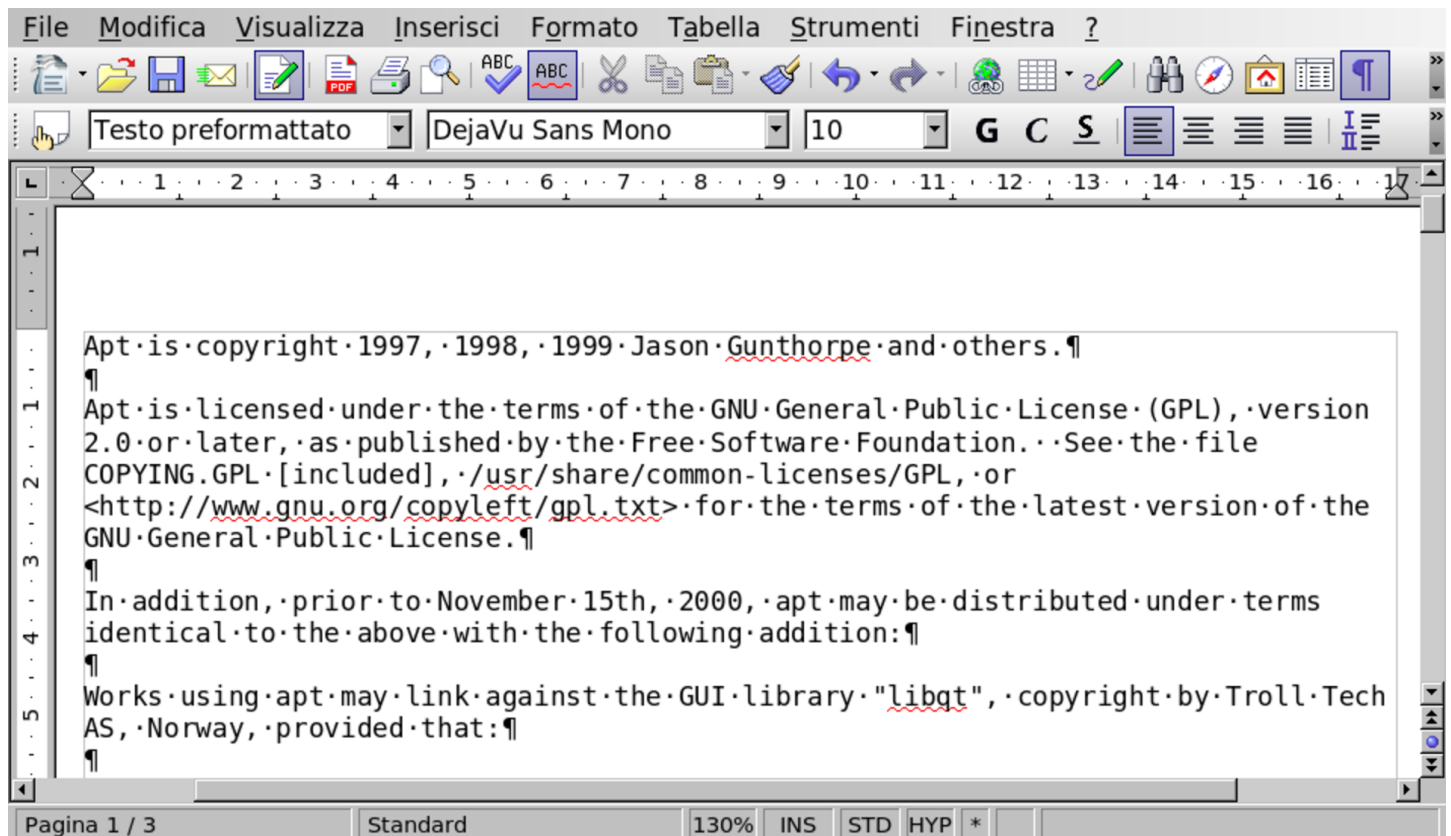
29.6.1 LibreOffice

LibreOffice²⁵ è un applicativo per l'automazione ufficio completo, che include il foglio elettronico, il programma di scrittura, il programma di disegno, il programma di presentazione e anche altri componenti. In generale è in grado di caricare e salvare anche in formati non propri, inclusi quelli di MS-Office.

Eventualmente, il capitolo 59 mostra l'uso di LibreOffice e di Abiword per le funzionalità principali che riguardano la scrittura.

Il programma di scrittura di LibreOffice è completo; oltre alla gestione degli stili, esiste anche la possibilità di fare la «stampa unione», ovvero la creazione di documenti con campi variabili, da completare in abbinamento a un elenco di valori (di solito per le lettere circolari).

Figura 29.40. LibreOffice per la scrittura di documenti.



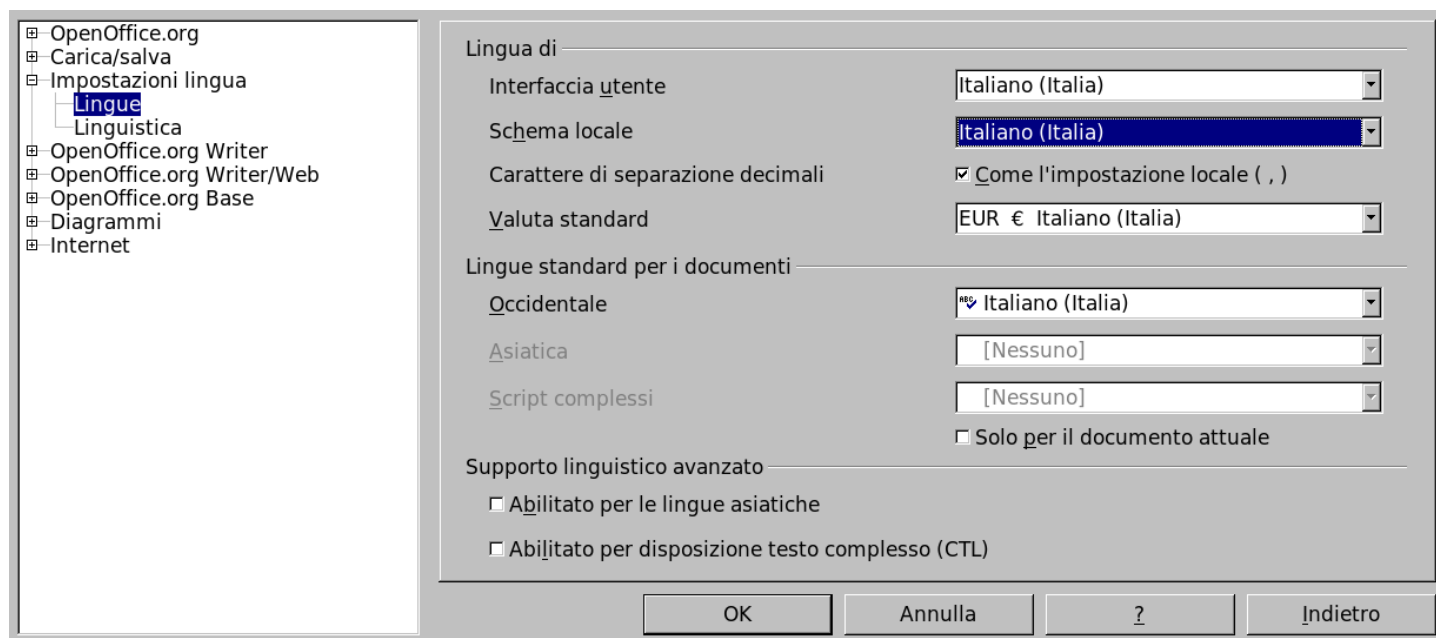
Anche il foglio elettronico di LibreOffice è completo, offrendo un alto grado di compatibilità con MS-Excel.

Figura 29.41. Il foglio elettronico di LibreOffice.

	E	F	G	H	I	J	K	L	M	N	O	P
7	04/12/2006	<i>Lu</i>	8.00	10.30	11.30	13.20			4.20	4.20		
8	05/12/2006	<i>Ma</i>							0.00	4.20		
9	06/12/2006	<i>Me</i>	8.50	13.20					4.30	8.50		
10	07/12/2006	<i>Gi</i>	9.40	13.20					3.40	12.30		
11	08/12/2006	<i>Ve</i>							0.00	12.30		

La configurazione di LibreOffice è accessibile dalla maschera che si vede nella figura 29.42. Si seleziona una voce nell'albero di sinistra, mentre a destra appare una maschera contestuale da compilare.

Figura 29.42. Configurazione di LibreOffice.



29.6.2 Gnumeric

<<

Gnumeric²⁶ è un applicativo per la gestione di fogli elettronici, realizzato in modo da essere funzionalmente equivalente a MS-Excel. Si tratta ovviamente di un'applicazione per X, che fa parte in particolare del lavoro generale su Gnome.

L'eseguibile da avviare è '**gnumeric**' e il suo aspetto si vede nella figura 29.43.

Figura 29.43. L'aspetto di Gnumeric con un foglio piuttosto complesso.

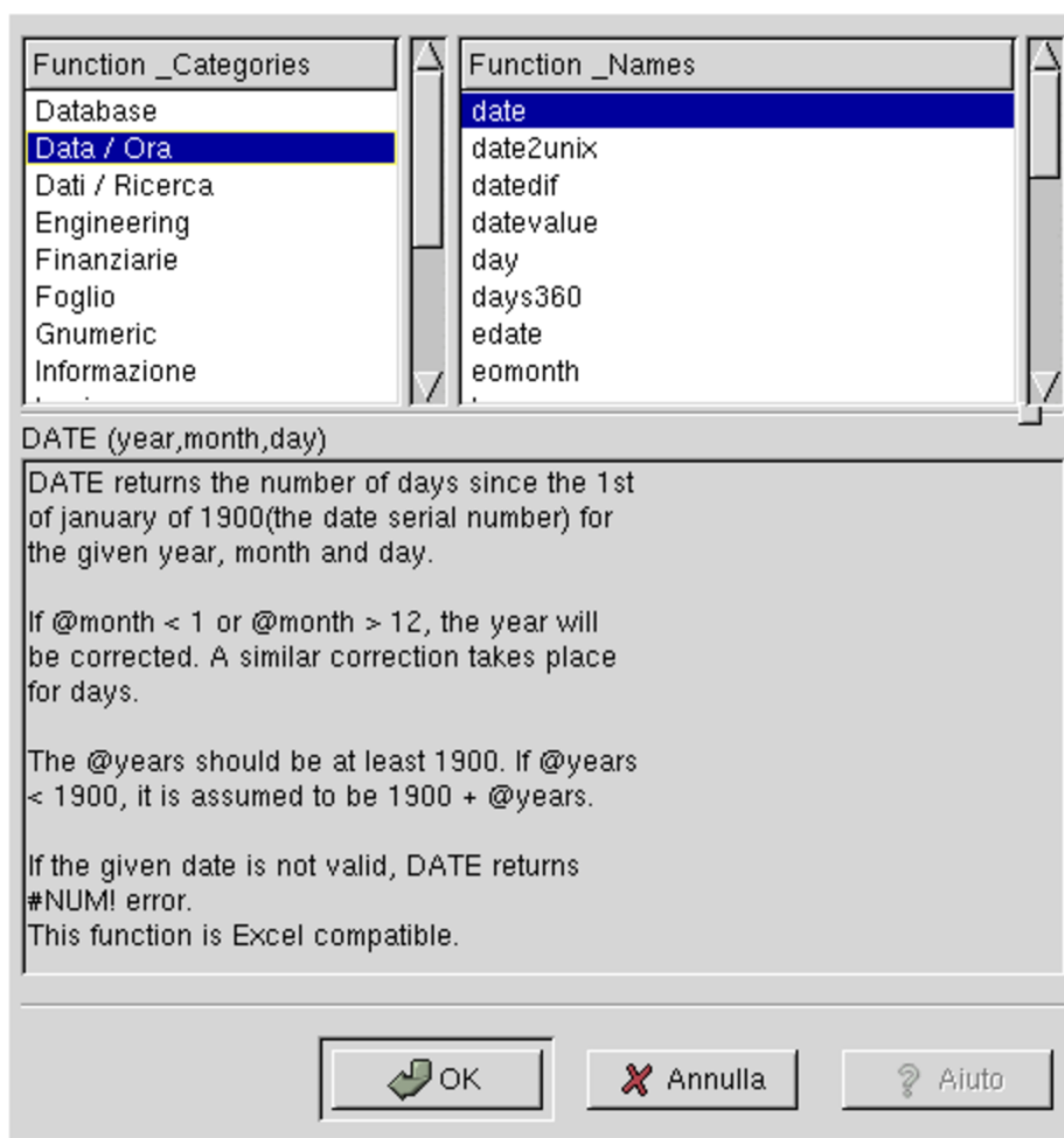
	F	G	H	I	J	K	L	M	N	O	P	Q
3		dalle	alle	dalle	alle	dalle	alle	ore giorn.	progr. sett.	Labor. e altri extra	CdC, Dip, ecc.	CdD
4	Lu							0:00	0:00			lezioni sospese
5	Ma							0:00	0:00			lezioni sospese
6	Me	8:00	13:25					5:25	5:25			
7	Gi							0:00	5:25			
8	Ve							0:00	5:25			
9	Sa							0:00	5:25			
10	DO							0:00	5:25			Domenica
11	Sr							0:00	0:00			

Orario di lavoro mensile

omma=0,225694444444444

Contrariamente ad altri fogli elettronici comuni, i nomi delle funzioni di Gnumeric rimangono in inglese. La figura 29.44 mostra la maschera che guida all'inserimento della funzione *date()*.

Figura 29.44. La maschera per la selezione guidata delle funzioni di Gnumeric.



29.6.3 AbiWord

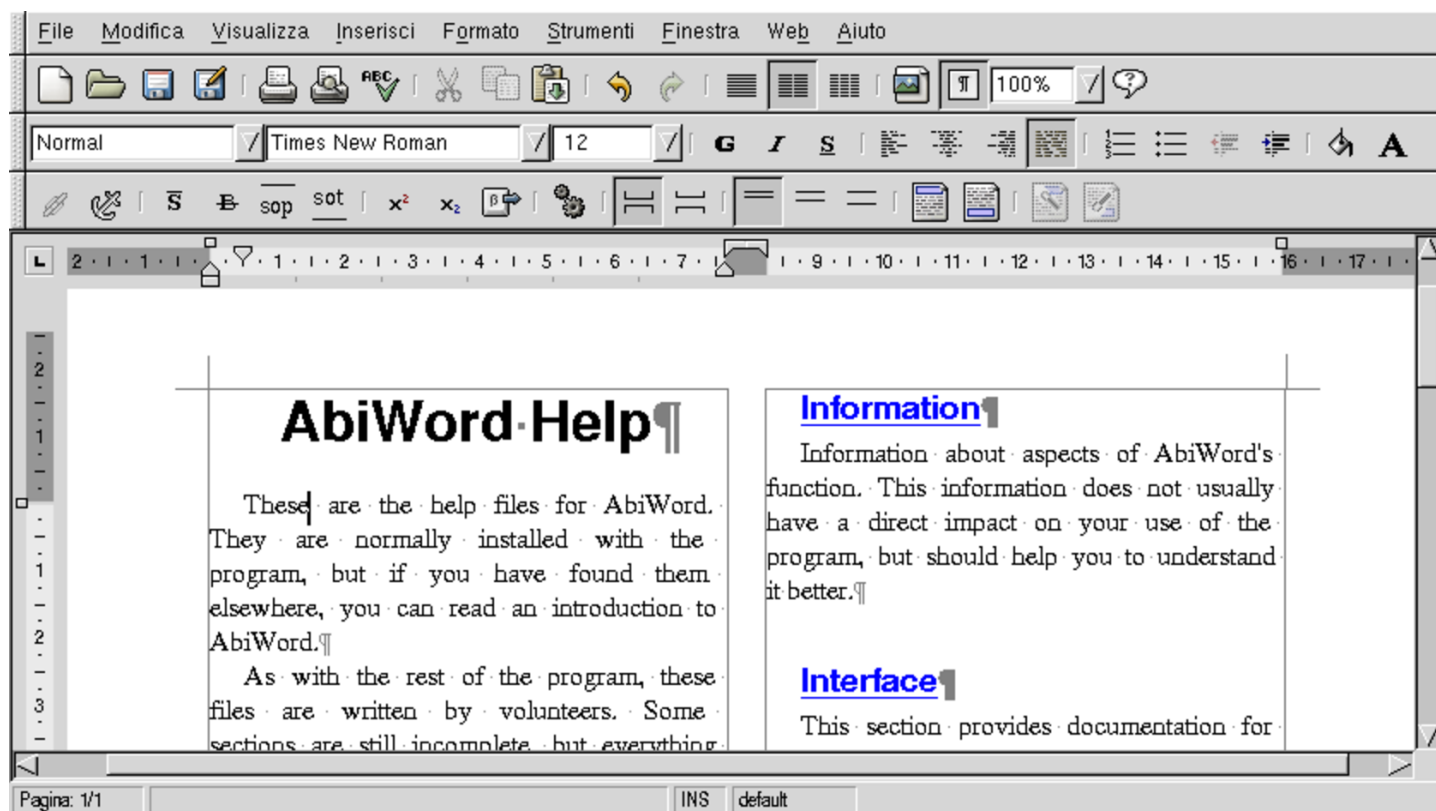
«

AbiWord²⁷ è un applicativo per la scrittura a composizione visuale (WYSIWYG), disponibile anche su piattaforme diverse dai sistemi Unix. Per quanto riguarda questi ultimi, si tratta ovviamente di un'applicazione per X, che utilizza in particolare le librerie GTK, cosa che lo integra esteticamente nell'ambiente di Gnome.

L'eseguibile da avviare è '**abiword**' e il suo aspetto si vede nella

figura 29.45.

Figura 29.45. L'aspetto di AbiWord.



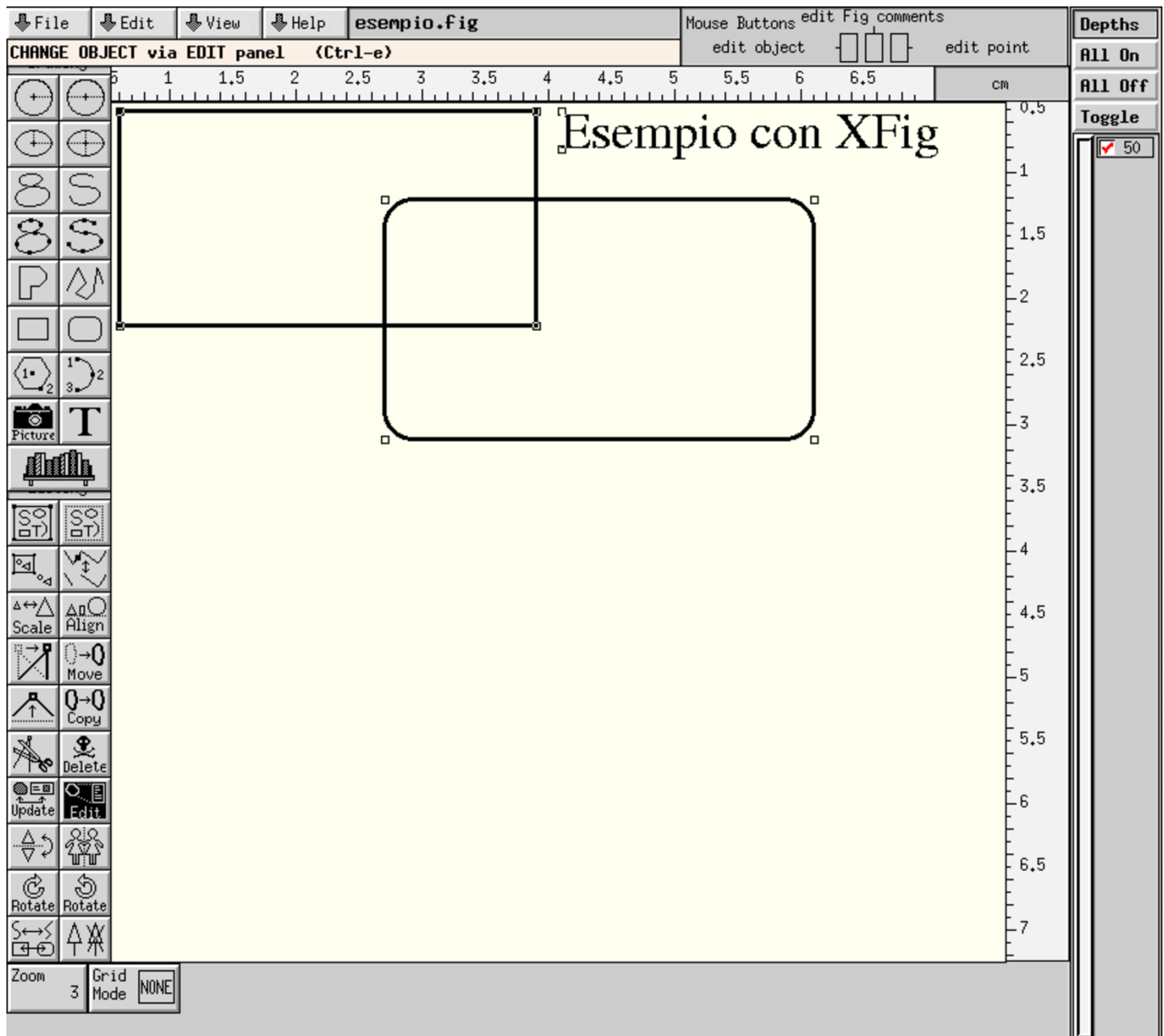
Eventualmente, il capitolo 59 mostra l'uso di Abiword e di LibreOffice per le funzionalità principali che riguardano la scrittura.

29.6.4 XFig

XFig²⁸ è un programma per il disegno vettoriale, con un'estetica un po' particolare, che differisce dagli applicativi comuni per l'automazione ufficio, in quanto si indica prima l'azione e poi l'oggetto a cui si applica.



Figura 29.46. XFig.



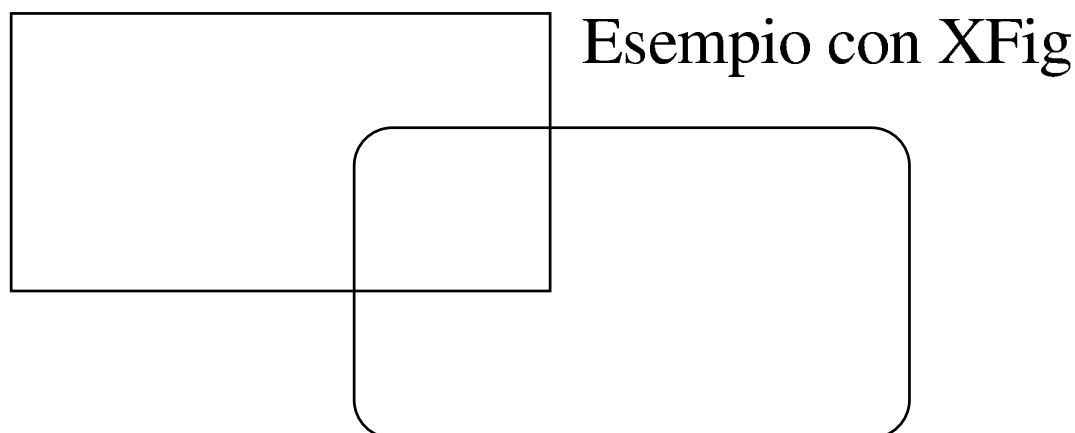
La descrizione che viene fatta in questo capitolo di XFig è molto superficiale, per consentire semplicemente di comprendere le possibilità di questo programma di disegno. La documentazione originale è comunque molto dettagliata e facile da consultare.

29.6.4.1 Formato FIG

XFig salva in un formato speciale: FIG. Si tratta di un file di testo normale, in cui si possono intuire i dati inseriti. Segue un esempio, che si traduce in pratica nella figura 29.48:

```
#FIG 3.2
Portrait
Center
Metric
A4
100.00
Single
-2
1200 2
2 2 0 1 0 7 50 0 -1 0.000 0 0 -1 0 0 5
      270 225 1755 225 1755 990 270 990 270 225
2 4 0 1 0 17 50 0 -1 0.000 0 0 7 0 0 5
      2745 1395 2745 540 1215 540 1215 1395 2745 1395
4 0 0 50 0 0 12 0.0000 4 180 1350 1845 360 Esempio con XFig\001
```

Figura 29.48. Esempio con XFig.



Si può osservare in particolare che il testo può essere rappresentato con caratteri normali e anche con sequenze ottali, nella for-

ma `\nnn`. Pertanto, la sequenza `\001` che si vede nell'esempio, rappresenta il codice `<SOH>`.

Un file FIG può contenere anche dei commenti, che appaiono preceduti dal simbolo `#`, nella prima colonna (all'inizio di una riga), a parte la prima direttiva del tipo `#FIG versione`, che serve a qualificare il contenuto del file.

29.6.4.2 Struttura generale dell'interfaccia grafica del programma

«

L'aspetto del programma non è quello comune ed è necessario apprendere il significato dei vari bottoni grafici e degli altri oggetti che stanno attorno alla superficie di disegno. In alto a sinistra si trovano alcuni pulsanti grafici che aprono dei menù a tendina:



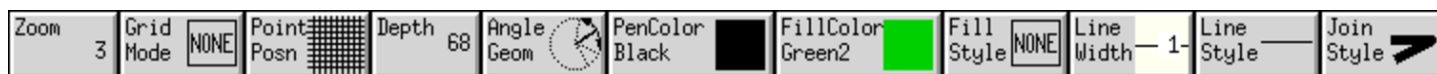
Sotto ai bottoni si trova la descrizione della modalità attiva (selezionata con uno dei bottoni disposti verticalmente sul lato sinistro), mentre un po' più a destra si vede il nome del file aperto. Più a destra, vengono riepilogate le funzionalità disponibili con il mouse, in base alla modalità attiva, oppure in base alla posizione del puntatore del mouse.

Sul lato sinistro è disposta una serie di pulsanti grafici: un primo gruppo riguarda l'inserimento di oggetti sulla superficie di disegno; il secondo consente la modifica degli oggetti. Selezionando uno di questi pulsanti grafici, se ne ottiene anche la descrizione nello spazio sotto i pulsanti del menù, dove in particolare viene anche descritto il modo per richiamare la funzione corrispondente attraverso la tastiera.

Sul lato destro, in alto, appaiono alcuni pulsanti grafici e una serie di caselle di selezione, riferiti alla gestione della profondità. Ogni oggetto che si colloca sulla superficie da disegno, ha un livello, definito come profondità. Un valore pari a zero rappresenta il livello più esterno, mentre un valore pari a 999 si riferisce a una profondità massima. Questa distinzione consente di sovrapporre gli oggetti stabilendo quale sia visibile e quale risulti nascosto: il livello di profondità più basso prende la precedenza e pone l'oggetto sopra gli altri. In questo modo, con il pannello dei pulsanti grafici, è possibile visualizzare solo uno o alcuni strati; nell'esempio mostrato, ne esiste uno soltanto, pari al valore 50:



Nel lato inferiore della finestra di XFig appaiono dei pulsanti grafici che variano a seconda del contesto e rappresentano delle opzioni che possono essere modificate:



29.6.4.3 Utilizzo comune del mouse

L'uso del mouse sulla superficie da disegno, non è intuitiva e ogni contesto ha la sua modalità di uso. Per quanto riguarda il disegno di oggetti comuni (poligoni, cerchi, linee), si possono riepilogare alcuni comportamenti importanti.

Il disegno di un oggetto richiede generalmente la pressione del primo tasto del mouse, in corrispondenza del primo punto che lo riguarda. Se l'oggetto prevede l'inserimento di più punti, questi si ottengono con la pressione successiva dello stesso tasto; se la quantità dei punti è indefinita, l'ultimo punto viene stabilito con la pressione del secondo tasto (quello centrale); al contrario, la pressione del terzo tasto quando l'oggetto non è terminato, provoca l'annullamento del suo inserimento.

Il disegno di un oggetto che prevede l'inserimento di più punti per il tratteggio di una linea, può essere iniziato e terminato con il secondo tasto del mouse, segnalando l'inizio e la fine; durante lo spostamento del mouse, si ottiene l'inserimento di diversi punti a intervalli relativamente frequenti.

In ogni caso, il promemoria che appare in alto a destra, descrive l'uso del mouse in base al contesto attivo; è sempre bene osservare questa guida per sapere come ci si deve comportare.

29.6.4.4 Esportazione e importazione di formati differenti

«

XFig è in grado di esportare un disegno in formati differenti, attraverso la voce *Export* del menù *File*, raggiungibile anche con la combinazione di tasti [*Meta x*] ([*Alt x*]). Al contrario, non è in grado di importare disegni da altri formati, ma per questo si possono usare programmi esterni.

Il primo programma da considerare è '**fig2dev**', in grado di convertire autonomamente un disegno FIG in altri formati (vettoriali o meno che siano):

```
fig2dev -L formato [opzioni] [file_fig [file_da_generare ] ]
```

‘**fig2dev**’ richiede un’opzione obbligatoria, ‘**-L**’, con la quale si specifica il formato che deve avere il file di destinazione, ottenuto dalla conversione; inoltre, se non si indicano i file, usa lo standard input per ottenere il file FIG da convertire e lo standard output per il file da generare.

Tabella 29.52. Utilizzo dell’opzione ‘**-L**’ con ‘**fig2dev**’.

Comando	Effetto
-L box	Genera il codice LaTeX necessario a ottenere una scatola della dimensione necessaria a contenere il disegno.
-L cgm	Genera un’immagine CGM (<i>Computer graphic metafile</i>).
-L latex -L epic -L eepic -L eepicemu	Genera un’immagine in formato LaTeX o in altri formati derivati più sofisticati.
-L gif	Genera un’immagine GIF.
-L ibmgl	Genera un disegno in formato IBM-GL, compatibile con HP-GL.
-L jpeg	Genera un’immagine JPEG.
-L map	Genera un file HTML contenente un riferimento a un’immagine esterna che però non viene creata automaticamente.

Comando	Effetto
-L mf	Genera un file da utilizzare con MetaFont per la costruzione di un insieme di caratteri.
-L mp -L nmp	Genera un file MetaPost o MultiMetaPost.
-L pcx	Genera un'immagine PCX.
-L eps -L ps -L pdf	Genera una trasformazione in EPS, PostScript e PDF.
-L pic -L tpic	Genera un file PIC, o TPIC, adatto per Groff.
-L pictex	Genera un formato PicTeX.
-L png	Genera un'immagine PNG.
-L ppm	Genera un'immagine PPM.
-L pstex -L pstex_i	Genera un formato PostScript senza testo e un formato LaTeX complementare, contenente solo il testo.
-L sld	Genera un formato SLD di AutoCAD.
-L textyl	Genera un formato Textyl.

Comando	Effetto
<code>-L tiff</code>	Genera un'immagine TIFF.
<code>-L tk</code>	Genera un formato adatto a TCL/Tk.
<code>-L xbp</code>	Genera un'immagine XBM.
<code>-L xpp</code>	Genera un'immagine XPM.

Oltre all'opzione `-L`, sono disponibili altre opzioni, alcune delle quali sono specifiche per il tipo di conversione che si intende attuare. Per l'elenco completo delle possibilità, basta consultare la pagina di manuale *fig2dev(1)*. Viene mostrato l'esempio della conversione di un file FIG in formato EPS:

```
$ fig2dev -L eps esempio.fig esempio.eps [Invio]
```

È interessante notare che il file che si ottiene include solo l'area del disegno che contiene qualcosa, indipendentemente dal formato della carta che può essere specificato nel preambolo del file FIG.

Può essere interessante anche l'uso del programma `pstoedit`, che è in grado di convertire un file PostScript in un disegno vettoriale; in particolare nel formato FIG:

```
$ pstoedit -f fig esempio.ps esempio.fig [Invio]
```

In questo esempio, si vuole convertire il file `esempio.ps` nel file `esempio.fig` (`pstoedit` è descritto nella sezione [26.1.7.1](#)).

Quando si esegue una conversione da un formato vettoriale finale, è molto probabile che si perdano delle informazioni significative sulla struttura del disegno. Per esempio, una linea tratteggiata contenuta

in un file PostScript può tradursi in un file FIG come tante piccole linee indipendenti.

29.7 Riferimenti

«

- *LibreOffice*, <http://www.libreoffice.org>
- Autori vari, *The Gnumeric manual*, <http://projects.gnome.org/gnumeric/doc/>
- *Xfig user manual*, <http://www.xfig.org/userman/>

¹ **X** MIT più altre licenze per porzioni particolari di codice

² **X** MIT più altre licenze per porzioni particolari di codice

³ **MLTerm** BSD e GNU GPL, a seconda della porzione di codice

⁴ **X** MIT più altre licenze per porzioni particolari di codice

⁵ **X** MIT più altre licenze per porzioni particolari di codice

⁶ **X** MIT più altre licenze per porzioni particolari di codice

⁷ **X** MIT più altre licenze per porzioni particolari di codice

⁸ **X** MIT più altre licenze per porzioni particolari di codice

⁹ **X** MIT più altre licenze per porzioni particolari di codice

¹⁰ **X** MIT più altre licenze per porzioni particolari di codice

¹¹ **X** MIT più altre licenze per porzioni particolari di codice

¹² **X** MIT più altre licenze per porzioni particolari di codice

¹³ **X** MIT più altre licenze per porzioni particolari di codice

¹⁴ **X** MIT più altre licenze per porzioni particolari di codice

¹⁵ **X** MIT più altre licenze per porzioni particolari di codice

- 16 Esistono almeno altri due programmi per visualizzare l'ora: **'oclock'** e **'rclock'**.
- 17 **X** MIT più altre licenze per porzioni particolari di codice
- 18 **XFE** GNU GPL
- 19 **Nautilus** GNU GPL
- 20 **Konqueror** GNU GPL
- 21 **File-roller** GNU GPL
- 22 **Putty** Software libero con licenza speciale
- 23 **Gmemusage** GNU GPL
- 24 **Etherape** GNU GPL
- 25 **LibreOffice** software libero: GNU GPL e altre licenze compatibili
- 26 **Gnumeric** GNU GPL
- 27 **AbiWord** GNU GPL
- 28 **XFig** software libero con licenza speciale

